AN ANALYSIS OF A DIGITAL POSITIONING SYSTEM

Yongyudha Nandidarbha

# NAVAL POSTGRADUATE SCHOOL

Monterey, California

# THESIS

AN ANALYSIS OF A DIGITAL POSITIONING SYSTEM

by

Yongyudha Nandidarbha

June 1975

Thesis Advisor: George J. Thaler

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

# REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

| | |
|---|---|
| 1. REPORT NUMBER | 2. GOVT ACCESSION NO. |
| 3. RECIPIENT'S CATALOG NUMBER | |
| 4. TITLE (and Subtitle) An Analysis of a Digital Positioning System | 5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1975 |
| | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s) Yongyudha Nandidarbha | 8. CONTRACT OR GRANT NUMBER(s) |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940 | 12. REPORT DATE June 1975 |
| | 13. NUMBER OF PAGES 132 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940 | 15. SECURITY CLASS. (of this report) Unclassified |
| | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital Positioning System

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Digital techniques can be effectively and economically used for control purpose. A digital positioning system is simulated, studied, and the results compared with the results from comparable analog system. Conventional compensation methods are used to compensate the comparable analog system and the results is used in simulation study of the digital system.

# An Analysis of a Digital Positioning System

by

Yongyudha Nandidarbha
First Lieutenant, Royal Thai Army

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1975

## ABSTRACT

Digital techniques can be effectively and economically used for control purpose. A digital positioning system is simulated, studied, and the results compared with the results from comparable analog system. Conventional compensation methods are used to compensate the comparable analog system and the results is used in simulation study of the digital system.

TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

Figure

## LIST OF SYMBOLS

| Symbol | Description of symbol |
|---|---|
| $B$ | Slope of ramp input function |
| $C$ | Analog output |
| $K_v$ | Servoamplifier gain |
| $N_t$ | Up-down counter output |
| $OP$ | Number of opening in a code disk |
| $R$ | Magnitude of analog step input |
| $R_n$ | Digital input |

ACKNOWLEDGEMENT

The author wishes to express his appreciation to his wife, Catherine, for her encouragement and assistance in preparing drawing and manuscript. The author also wishes to acknowledge the generous assistance given by Mr. Martin Dost in advising on the DSL/360 Language and especially to Dr. George J. Thaler for originally suggesting the problems and for his continuous encouragement and guidance throughout this research.

# I. INTRODUCTION

Control system complexity and demands on reliability are rising with ever increasing automation. More and more control system engineers are looking to solid-state digital techniques for new answers to the old problems of reliability, complexity, and economy. Digital techniques provide many advantages such as accuracy and computational speed as well as flexibility and versatility, which analog techniques could not provide. By the use of digital techniques in control systems, sensitive, and efficient control devices can easily be built. Another significant advantage of digital control systems is that the utilizing of digital components in a control system allows time sharing of important parts of the system which results in economy in the use of equipment. The digital elements also require much less power than the analog elements. The data signals in digital control systems can also be transmitted and received in pulse-code form almost error-free, even in transmission through a noisy media. Furthermore, a digital computer in control gives another important adventage, it makes feasible system compensation by nonlinear programming and adaptive or self-optimizing control.

The digital servo system is becoming widely accepted in machine tool control, radar systems, navigational system, and many other fields. A digital servomechanism uses numerical information to provide a simple but highly accurate method of positioning control. Figure 1 shows a block diagram of a typical digital servo system. The major components of the basic digital

servo system are :

1. Controlled system
2. Control elements
3. Analog-to-Digital converter
4. Digital-to-Analog converter
5. Digital data processing equipment or computer

The controlled system is normally a linear servomotor but step motors have been used for quite some time in both closed loop and open loop digital positioning systems. The digital computer is used as the central control unit for data processing, system compensation, and decision making in some cases. In a simple digital servo system, a comparator such as an up-down counter can be used. The digital- to-analog and analog-to-digital converters are used to interconnect the digital and analog quantities for controlled and control elements. One familier type of analog-to-digital converter-encoder is the shaft-position encoder disk employing a gray or binary code. This digitizer is used to translate the shaft position into digital notation by making use of a mechanically rotating shaft with coding disk centered on the shaft. Figure 2 illustrates a typical coding disk for a shaft digitizer. The brushes contact the disk along the reading line at the point shown with a single brush positioned in the center of each concentric band. These concentric rings represent $2^0$, $2^1$, $2^2,\ldots,2^n$, and each ring is constructed of several segments made of either conducting material (the darkened areas) or insulating material ( the unshaded areas). A signal is connected to the

conductor, and if a given brush makes contact with a segment of conducting material, a " 1 " signal will result, but if the brush is over the insulating area, the output from the brush will be a " 0 ". The resolution of this encoder is determined by the number of channels. For a 4-channel coding disk, a resolution of one part in sixteen is achieved. Two types of encoder disk, the gray and binary code disks are shown in Figure 3. The coded disk may be made of transparent and opaque materials arranged in coded patterns. The photocells and sources are used to read the angular position in a digital code.

In operation, the analog-to-digital converter converts the analog output of the controlled system into a numerical quantity, which is compared with a numerical order, and the difference is converted by the digital-to-analog converter into an analog signal capable of driving a servomotor until the difference is eliminated.

A. OBJECTIVES

The methods for analysis and synthesis of the analog system such as Bode plots, Root Locus, Parameter plane plots, etc. have been established for a long time. These methods work very well, however, for the discrete or digital system, there is no such method that is as good as those for the analog system. Even the well known z-transform method can become very complicated if the order of the systems analyzed are higher than third or fourth order.

The objective of this paper was to investigate a typical

digital positioning servo system in order to obtain some good methods for analysis and synthesis. In the proposed system, light travels from a light source through equally spaced openings in the code disk to activate the sensor. The sensor, in turn, generates feedback pulses at equilibrium angular position. The sampling period of this system is not constant, depending on the velocity of the motor, the z-transform method of analysis and design cannot be used here. Another objective was to find a suitable mathematic representation of such a system.

All of the studies had been done by simulation using Digital Simulation language DSL/360 on the IBM 360 computer system, this language made the simulation quite easy, however, care must be taken in order to obtain accurate integration in the system with high velocity.

In Chapter II, the system identification and method of simulation are given. Chapter III presents the discussion of the results obtained from the simulation of second and third order servo systems. This includes the study of the stability limit, percent overshoot, and the effects of variation of the servoamplifier gain. This chapter also shows the results obtained from investigating the behavior of the system with ramp input. The Bode plot and parameter plane plot were used to compensate the analog system in Chapter IV. The results were then used in simulating the digital system, since there was no direct compensation method.

The results were obtained both in graphs and printed output,

however, only graphs are included. Listing of all computer programs are presented after the conclusion and recommendation in Chapter V.

In each study, the digital system with several numbers of opening slots and the continuous counterpart were simulated. The studies of the digital systems were based on the results of studies of the continuous system.

# II. THE DIGITAL POSITIONING SERVO

## A. PHYSICAL PROPERTIES

The digital positioning servo system under consideration consists of a digital comparator, a digital-to-analog converter, a controlled system, and an analog-to-digital converter. The analog-to-digital converter consists of a low inertia encoder disk mounted tightly on the servomotor shaft, two photoelectric sensor units, and a direction-sensing logic circuit. The encoder disk has uniformly spaced slots around its circumference. Each photoelectric sensor unit consists of a light source and a photodiode. Figure 4 illustrates the encoder disk and the sensor units and Figure 5 illustrates the direction-sensing circuit.

Each time an opening passes through the beam from the light source, a pulse is generated by the sensing unit, and it will eventually be fed to the up-down counter which acts as the digital comparator. The second sensor unit is for the purpose of direction sensing. When the servomotor turns in forward direction, the feedback pulse becomes a down-enable signal, when the servomotor turns backward, the feedback pulse becomes an up-enable signal. The two sensors are adjusted so that their respective outputs, X and Y, overlap as shown in Figure 6. Figure 6-A and 6-B show the inputs and outputs of the direction-sensing circuit when the servomotor turns forward and backward, respectively. Figure 7 shows a typical digital-to-analog converter.

For an input as a step function, an appropriate binary number representing the size of the step function is fed into the up-down

counter. The output of the up-down counter is, in turn, converted into an analog signal and is sent to the servoamplifier to drive the servomotor. The sensing device will measure the shaft position, encode, and send this information to the up-down counter. The positioning process will be finished when the binary number in the up-down counter becomes zero.

B. COMPUTER SIMULATION

The DSL/360 language was specially designed for use in simulation. DSL/360 provides many advantages over other languages and subroutines in existence, especially in simplicity and time saving. DSL/360 not only provides these advantages in simulating the linear physical system, but in simulation of the non-linear system also. DSL/360 provides a basic set of function blocks from which a physical system may be modeled such as integrators, quantizer, pulse generators, function generators, and so on. In addition, FORTRAN Library functions from the Scientific Subroutine Package may be utilized. The user may also provide his own subroutines if necessary.

In programming, first, each part of the digital positioning servo system was separated into single function blocks as shown in Figure 8. Next, each single function block was replaced by a related DSL function or standard FORTRAN Library function as appropriate. Finally, all the DSL and FORTRAN Library function were put together in the DERIVATIVE REGION of the program, One advantage obtained in using the DERIVATIVE REGION is that the functions are

not required to be in any specific order.

In the early parts of simulation. The servomotor was assumed to be a second order motor with the Transfer function

$$G_m(s) = \frac{K_v}{s(s + 1)} .$$

Later, the third order servomotor with the transfer function

$$G_m(s) = \frac{K_v}{s(s + 1)(0.2s + 1)}$$

was used. The FORTRAN Library function

$$Y = AINT(X)$$

was used to modelled the quantizer. This function was also used to quantize the ramp input function. The gain of the digital-to-analog converter was designed to be equal to $1/K_c$ to make the system a unity feedback system.

The fourth order Runge-Kutta with fixed integration interval (INTEG RKSFX) Integration routine was used in all simulations to assure the accuracy of the results. It is very important to choose a correct integration interval (DELT), since too large DELT will cause inaccurate integration and too small DELT will result in excessive computing time. The DELT used in each simulation is approximately five to ten percent of the time interval between two consecutive pulses generated by the digitizer at its maximum angular velocity in that specific simulation.

In multiple runs, the TERMINAL region was used to terminate the previous run, to provide the new sets of parameter values,

and to terminate the program after the output data of the last run had been plotted. The FORTRAN subroutine DRWG was used in the SAMPLE region to collect points for each sample interval (DELS) to be plotted by the CALCOMP XY Plotter.

In most cases, The digital system with 10, 45, 90, and 180 openings and the continuous system were simulated and the results were compared. A system with one opening was used in stability limit studies since it was the worst case. To compare the outputs of the digital system with output of the continuous system, the inputs of all systems must be equal. If a step input to the continuous system is 1.25664, inputs to the digital system with 10, 45, 90, and 180 openings must be 2, 9, 18, and 36 respectively.

These input values can be readily calculated by the method shown in example below.

EXAMPLE

To calculate comparable step inputs to the continuous and digital system with 10 openings.

From Figure 8,

$$V_{in(continuous)} = R - C$$

$$V_{in(digital)} = \frac{R_n - IGBIT}{K_c}$$

$$= \frac{2\ (R_n - IGBIT)}{OP}$$

for zero initial conditions,

$$V_{in(continuous)} = R$$

$$V_{in(digital)} = \frac{2\ R_n}{OP}$$

to have equal inputs to servomotors, requires

$$R = \frac{2\ R_n}{OP}$$

by arbitarily choosing $R_n = 2$

$$R = 1.25664$$

The values of inputs to digital system with other numbers of opening were also based on this selection.

# III. DISCUSSION

The first part of this chapter contains a discussion of the response of the second order servo system to a step function input. The third order system with both step and ramp function input is discussed in the later part of the chapter.

## A. STEP RESPONSE OF THE SECOND ORDER SYSTEM

The main concern in the response of the second order servo system was its stability limit. The forward gain of the system had been increased until the stability limit was obtained.

The model of the second order continuous system used in this simulation is always stable regardless of the size of the gain. The digital system was also stable at all value of gain regardless of the number of openings in the position encoder. The gain of the system was increased to $10^7$ to make sure that the system would always be stable. Figure 9, 10, 11, 12, 13, and 14 show the step response and plots of the binary numbers in the up-down counter versus time for the digital system with number of openings equal to 1, 10, 45, 90, and 180, and for the continuous system, respectively. Figure 15, 16, 17, 18, 19, and 20 show their respective phase plane plots. The plots of the output of the up-down counter were not scaled correctly, however, they could be read easily. Each quantizied step of the $N_t$ plot represents a one.

The difference that can readily be seen in the responses of the digital and continuous systems is the magnitude of both the angular displacement and velocity. The difference is very pronounced

when the number of openings is small. The digital system with fewer openings has longer time delay between two consecutive feedback pulses. This time delay has allowed the motor to move farther with higher velocity, similar to the response of the system without a feedback loop, before the excitation signal is reduced by the next feedback pulse. The system with a large number of openings has a very high sampling rate, therefore this system behavior is much closer to that of the continuous system. a code disk with 20000 openings has been developed. This disk gives the system almost the same response as that of a continuous system.

The phase plane plots of the digital system are not as smooth as those of the continuous system. They actually consist of many short sections. Each portion represents the delay interval between two consecutive pulses. This can easily be observed in the phase plane plot of the system with lower number of openings such as shown in Figure 16.

Figure 21 contains the comparison of the angular displacement of the analog and digital system with 10 and 45 openings. This plot clearly shows that the natural frequency of this second order digital system is smaller than that of the analog system. It indicates that the root locus of the digital system would cross the $\omega$-axis closer to the origin than the root locus of the continuous system. This distance becomes smaller if the number of openings is decreased.

The plots of the step response of the digital system also consist of many short sections. It is very easy to observe this phenomenon in the response of the system with very low gain such as those shown in Figure 22 and 23. Figure 24 demonstrates how this phenomenon occurs. At the beginning of the simulation, the system behaves like a system without a feedback loop. The rate of increase of velocity is very high at first, however, it becomes lower with time. The displacement plot also becomes more linear with time. When the displacement is large enough, a feedback pulse is generated, the output of the up-down counter is changed by one, the velocity and the rate of change of the displacement are changed to follow the new input voltage to the servoamplifier. This process occurs repeatly until the output of the up-down counter is eliminated, and the system eventually comes to rest.

Figure 22 and 23 also illustrate that the digital system behaves like the continuous system in another aspect. When the gain is reduced, the peak overshoot decreases. This peak overshoot can be eliminated by making the gain small enough. It is also observed that, for all cases studied, the steady state displacement of the digital system is always larger than that of the continuous system with comparable size of step input. The difference, however, has never exceeded the angular displacement between two adjacent openings. The system with more openings in the encoder disk gives smaller error than the system with fewer openings. The maximum error for the system with 10, 45, 90, and 180 openings are less than 0.6283, 0.1396, 0.0698, and 0.0349 radians respectively.

## B. THIRD ORDER SYSTEM

A third order system was studied in greater detail than the second order system. In addition to what had been studied from the second order system characteristics, the variation of the percent overshoot was investigated in this part.

### 1. Stability limit

A third order continuous system with the transfer function shown in Chapter II reaches the stability limit when the servoamplifier gain, $K_v$ is equal to 6. The digital systems, however, require larger gains to reach their stability limits. Figure 25, 26, 27, 28, and 29 illustrate the phase plane plots of the third order digital system with the gain equal to 6 and the number of openings equal to 1, 10, 45, 90, and 180 respectively. Figure 30 shows the phase plane plot of the continuous system with the same gain. The phase plane plots of the third order digital system converge into a limit cycle and can clearly be observed in the case the system has 1 or 10 openings. By inspecting the characteristic of the phase plane plots, especially in the case where the number of openings is small, it is obvious that the third order digital system requires higher gain to reach the stability limit than the continuous system. The time delay also causes the displacement, velocity, and limit cycle size of the digital system to be larger than those of the continuous system.

Since there had not been any known method that can be used to find the gain at stability limit of this digital servo

system, a trial and error method was used to find this gain, the gain of the third order digital system with 10 openings at its stability limit was found to be approximately 6.25. This gain would be reduced toward 6 if the number of openings is increased. The size of the limit cycle is also decreased with the increase of the number of openings.

The results from simulating the third order digital servo system with number of opening equal to 10 as illustrated in Figure 31, 26, 32, and 33 also show that the size of the limit cycle increases with the gain. The increase of the gain also causes the system to reach this limit cycle faster. Upon making the gain greater than the stability limit gain, The phase plane plot diverges for a period of time before reaching another limit cycle as shown in Figure 34. For higher gain, The phase plane plot diverges into a limit cycle and stays in the limit cycle for a period of time then diverges to infinity as shown in Figure 35. With the gain a little higher as shown in Figure 36, the phase plane plot diverges to infinity but has a tendency to go into a limit cycle at first. When the gain is much higher, the system just becomes unstable as shown in Figure 37.

2. Percent overshoot

For a continuous and stable servo system, the overshoots of the step response always decrease with time, and the plot of the log of the percent overshoot versus time is a straight line. The step response and the plot of log of percent overshoot of the continuous system are shown in Figure 38 and 43 respectively.

To investigate the behavior of the overshoot of the digital positoning servo system, a gain of 2.5 was used. With this gain, $\zeta$ and $\omega_n$ of the continuous system are equal to 0.166 and 1.54 respectively. Figure 39, 40, 41, and 42 illustrate the step response and output of the up-down counter of the third order digital servo system with 180, 90, 45, and 10 openings respectively. The step response of the continuous system shows 6 prominent overshoots, the responses of the digital system with 180, 90, 45, and 10 openings show 5, 4, 3, and 2 overshoots respectively. The time at which each of the corresponding peaks of all systems occur is the same. Figure 43 illustrates the plots of the percent overshoot of the responses of the continuous and digital system with 180, 90, 45, and 10 openings versus time. The numerical values of percent overshoots are included in table I. The inputs to all systems were adjusted so that they would be equal and the percent overshoot could be calculated from the same basis, the steady state output of the continuous system.

The overshoots of all digital systems are higher than the corresponding peaks of the continuous system. The difference, however, does not exceed the angle of the two adjacent openings. The digital system with smaller number of openings has larger percent overshoot than the system with larger number of openings. The steady state response of the digital system is also larger than that of the continuous system. The difference is also less than the angular displacement between two adjacent openings.

At the beginning, the plot of the percent overshoot of the

digital system tends to be parallel to that of the continuous system. The plot of the percent overshoot then diverges to the smallest peak which is slightly larger than the angular displacement between two adjacent openings. Since the system with fewer openings has larger displacement between the openings, its response requires less time to reach the last overshoot. The number of its overshoots is also less than that of the system with more openings.

The responses of all the digital systems that have less than 10 openings were also studied. All of the responses have two overshoots and the characteristics are similar to those discussed above. However, only the response of the system with one opening, which is the extreme case, is illustrated in Figure 44.

### 3. Response to ramp input

The ramp input function, Bt, was also applied to the third order digital positioning servo system in order to investigate its stability. The servoamplifier gain used in this study was 2.5 and the slope of the ramp was varied from 0.5 to .5. Practically, a method of applying a ramp input to the digital servo system is to apply a series of impulses into the up-down counter. The slope of the ramp can be controlled by varying the period of the impulse. A digitized ramp function was used in this simulation instead of the impulses. The study of the response of the digital servo system with ramp input was done by simulating the system with 10 openings only. It had been shown from the previous studies that the digital system with more openings behaves closer to the continuous system than the system with fewer openings.

Figure 45 and 46 illustrate the response of the digital system with 10 openings and the slope of the ramp input equal to 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, and 5.0. When the slope is small, for example B = 0.5, the rate of the input pulse applied to the up-down counter is slow and causes the velocity of the shaft and the feedback sampling rate to be too slow and nonuniform. The nonuniform and slow rate of output of the up-down counter, in turn, cause the shaft to turn abruptly and the system becomes unstable. The longer time delay is also an important cause for the instability of the system. The plot of velocity and output of the up-down counter for the digital system with the slope of the ramp input to the system equal to 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, and 5.0 are shown in Figure 47, 48, 49, 50, 51, 52, 53, and 54 respectively. When the slope of the ramp input is increased, the input pulses are fed to the up-down counter at higher rate, the servo shaft turns faster, the feedback sampling rate becomes higher, the output of the up-down counter becomes more linear and the velocity of the shaft becomes more stable. Unlike the continuous system, the velocity of the digital system will not be constant no matter how large a slope is used for the ramp input. It rather varies within a small limit. This limit cycle is caused by the time delay between pulses. The limit cycle will be smaller if the time delay is shorter.

Another interesting result observed is that the rate of decrease of the magnitude of the limit cycle with the increase of the slope of the ramp is nonuniform. At first the magnitude

of the limit cycle becomes very small when the slope is increased to 2.5. However, when the slope is increased to 3 and 4, the magnitudes of the limit cycles are some what larger. The limit cycle is reduced to a smaller size again when the slope is increased to 5.

## IV. SYSTEM COMPENSATION

Since there had not been any known method that can be used to compensate this digital positioning servo system directly, the author chose to compensate the comparable continuous system with known methods and simulated the compensated system as both a continuous and a digital system. The results from the simulation were then compared and studied. The continuous and the digital system with 10, 45, and 180 openings were simulated and their step responses were plotted on the same graph to make the comparison more obvious. The simulation of the digital system with 90 openings was omitted to prevent the graph from being overcrowded. The system compensated was the third order servo with servoamplifier gain equal to 6.

### A. LAG COMPENSATION

Since the programs and computer facility were available, the parameter plane plot method was used in lag-compensating of the system. This method is very efficient and flexible and the computer make it very easy to obtain the plot. The author selected the PARAM M program to use in this compensation. The complete details of this program are shown in the appendix.

In order to obtain the parameter plane curve, the characteristic equation of the system and other requirements such as the value of $\zeta$, $\omega_n$, $\omega$, or $\sigma$ needed to be plotted and the scales of the plot must be provided as the input data to the PARAM M program. Figure 55 illustrates the block diagram of the third order

continuous system with the lag compensation section. The lag compensation block has the transfer function

$$G_m(s) = \frac{(\tau_z s + 1)}{(\tau_p s + 1)}$$

The transfer function of the third order system with the lag compensation becomes

$$G_m(s) = \frac{(\tau_z s + 1)}{(\tau_p s + 1)} \cdot \frac{6}{s(s + 1)(0.2s + 1)}$$

or

$$G(s) = \frac{p}{z} \frac{(s + z)}{(s + p)} \cdot \frac{30}{s(s + 1)(s + 5)}$$

The characteristic equation of this system is

$$s^4 + (6 + p)s^3 + (5 + 6p)s^2 + (5p + 30\tfrac{p}{z})s + 30p = 0$$

let

$$\alpha = p$$

$$\beta = \frac{p}{z}$$

the characteristic equation becomes

$$s^4 + (6 + \alpha)s^3 + (5 + 6\alpha)s^2 + (5 + 30\beta)s + 30\alpha = 0$$

The parameter plane plot of this characteristic equation is shown in Figure 56. $\alpha$ and $\beta$ are the unknown pair of variables which their values can be directly selected from the suitable value of $\zeta$ and $\omega_n$. The values of pole and zero can also be readily obtained from the values of $\alpha$ and $\beta$. Three pairs of poles and zeroes were chosen and simulated. The step responses of the compensated continuous and digital system with 10, 45, and 180

openings are illustrated in Figure 57, 58, and 59. The values of pole, zero, $\zeta$, $\omega_n$, and predicted and resulted values of $M_{pt}$, $t_p$, and $t_s$ of the compensated system are included in table II.

The simulated results show that the characteristics of the digital system are similar to those of the continuous system. The results also agree with those studied in the previous chapter. The magnitude of the peak overshoot and steady state response of the digital system are always larger than those of continuous system. All the differences are also within the limit of the angular displacement between two adjacent openings.

Care must be taken in selecting the value of the zero for the lag compensation section. If this zero was too close to the origin, the real root would dominate the system instead of the selected pair of imaginary roots. If the zero was too far from the origin, the value of $\zeta$ would be too small and cause the system to be very slow.

## B. LEAD COMPENSATION

The same characteristic equation was used to obtain the parameter plane plot for lead compensation. However, the scales, the starting value of $\omega_n$, the required values of $\zeta$ and $\omega_n$ curves, and etc. must be reselected in order to move the curves into required region for the lead compensation. The block diagram of the system with lead compensation section is the same as that of the lag compensated system. Its parameter plane plot is shown in Figure 60.

The pole and zero of the lead compensation section were kept

within one decade apart to prevent the severe reduction of the forward transfer gain of the section. This imposes another limitation into the selection of the value of $\beta$. The value of $\beta$ should not be larger than 10. Three pairs of poles and zeroes were selected and simulated. Figure 61, 62, and 63 illustrate the step response of the compensated system in both continuous and digital versions. Table III consists of the values of pole, zero, $\zeta$, $\omega_n$, predicted and resulted values of $t_p$, $M_{pt}$, and $t_s$ of the compensated system. Since the natural frequency of the lead compensated system is larger than that of the lag compensated system it results in smaller $t_p$ and $t_s$. The other results of this study are similar to those described in the lag compensation study above.

## C. LAG-LEAD COMPENSATION

The block diagram of the third order continuous system with lag and lead compensation sections is shown in Figure 64. There are four variables in this system which make it more difficult to use the parameter plane plot method in compensation. The Bode plot method was used in this compensation. This method is rather simple but requires some degree of neatness. Figure 65 and 66 illustrate the Bode plots of two different lag-lead compensations. The limitation in selecting the values of the zero of the lag section and the spread between the pole and the zero of the lead section are also imposed here. Both of the compensated systems were also simulated and studied. The results of the simulation of the lag-lead compensated systems are shown in Figure 67 and 68.

Table IV consists of the values of pole, zero, $\zeta$, $\omega_n$, predicted and resulted value of $t_p$, $M_{pt}$, and $t_s$. The results of the lag-lead compensation study are also similar to those of lag and lead compensation studies.

D. COMPENSATION DISCUSSION

The results from all compensation studies as well as those from the previous chapter show that the steady state response of the digital system is always within the angular displacement between two adjacent openings larger than that of the comparable analog system. However, this difference is not constant, it changes randomly within the limit. The exact position of the steady state response of the system can not be determined. It seems that the best method to compensate the digital system is to compensate its analog counterpart.

The maximum error of the system under investication is found to be

$$\frac{1.296 \times 10^6}{OP} \quad \text{seconds}$$

In comparison, the analog system has the accuracy of approximately 2 minutes of arc and the digital system with gray or binary shaft-position encoder has the accuracy of

$$\frac{1.296 \times 10^6}{2^n} \quad \text{seconds}$$

where n is the number of the conducting channel (see ref. 5). In order to have the same accuracy as of the analog system, this digital system must have approximately 10000 openings in the encoder

disk. The digital system with gray or binary code disk requires 13 channels in the code disk to acquire comparable accuracy. It is trivial that this digital system has an economic advantage over the gray or binary code disk system. This system requires only two sensing units for any number of openings. The gray or binary shaft-position encoder unit requires one sensing device for each channel in the code disk and the sensing circuit has to be more complicated.

Another adventage is that all the studies show that the response of this digital system is always larger than the response of the continuous system. This observation could be used to reduced the steady state error by making the size of the step input to be less than the required input to the system. The difference should be as close to a half of the maximum error as possible. The difference of exactly a half of the angular displacement between two adjacent openings would reduced the steady state error to within one half of the maximum error.

# CONCLUSSION

The characteristics of the digital positioning system under investigation can be very similar to the analog system if the feedback sampling rate is high enough. This system has some characteristics which are superior to the conventional analog and sampled-data system, especially in accuracy and economy.

Compensation of this system can be accomplished by using the results obtained from compensating its analog counterpart by conventional methods. The accuracy can be readily improved by modifying the encoder disk.

From the discussion above, it can be seen that there are many areas left to be studied. The author believes that further study of the mathematical analysis and of the direct method of compensation are necessary.

TABLE I

PERCENT OVERSHOOT

| | percent 1st overshoot | percent 2nd overshoot | percent 3rd overshoot | percent 4th overshoot | percent 5th overshoot | percent 6th overshoot |
|---|---|---|---|---|---|---|
| analog | 56.97 | 19.98 | 7.01 | 2.46 | 0.86 | 0.30 |
| digital | | | | | | |
| CP=180 | 59.20 | 21.20 | 8.45 | 3.70 | 2.81 | - |
| OP=90 | 61.18 | 23.19 | 10.00 | 6.00 | - | - |
| OP=45 | 65.50 | 26.36 | 13.48 | - | - | - |
| OP=10 | 94.50 | 54.50 | - | - | - | - |

TABLE II

LAG COMPENSATED THIRD ORDER SYSTEM

| pole | zero | $\zeta$ | $\omega_n$ | predicted results | | | simulated results | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | $t_p$ | $M_{pt}$ | $t_s$ | $t_p$ | $M_{pt}$ | $t_s$ |
| 0.013 | 0.18 | 0.36 | 0.99 | 3.39 | 1.31 | 11.20 | 6.40 | 1.37 | 12.8 |
| 0.017 | 0.19 | 0.34 | 1.01 | 3.29 | 1.33 | 11.64 | 5.70 | 1.40 | 12.5 |
| 0.01 | 0.15 | 0.4 | 0.87 | 4.00 | 1.26 | 11.49 | 7.35 | 1.32 | 15.8 |

TABLE III

LEAD COMPENSATED THIRD ORDER SYSTEM

| pole | zero | $\zeta$ | $\omega_n$ | predicted results | | | simulated results | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | $t_p$ | $M_{pt}$ | $t_s$ | $t_p$ | $M_{pt}$ | $t_s$ |
| 14.5 | 1.85 | 0.35 | 3.5 | 0.94 | 1.32 | 3.27 | 0.97 | 1.4 | 3.17 |
| 15 | 1.5 | 0.35 | 4.0 | 0.83 | 1.32 | 2.86 | 0.85 | 1.37 | 2.75 |
| 19 | 1.9 | 0.37 | 3.5 | 0.96 | 1.3 | 3.09 | 0.90 | 1.37 | 3.12 |

TABLE IV

LAG-LEAD COMPENSATED THIRD ORDER SYSTEM

| lag section | | lead section | | | | predicted results | | | simulated results | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| pole | zero | pole | zero | $\zeta$ | $\omega_n$ | $t_p$ | $M_{pt}$ | $t_s$ | $t_p$ | $M_{pt}$ | $t_s$ |
| 0.25 | 0.5 | 20.0 | 2.0 | 0.34 | 2.27 | 1.61 | 1.33 | 5.18 | 1.46 | 1.36 | 4.05 |
| 0.2 | 0.5 | 12.5 | 1.25 | 0.39 | 2.6 | 1.44 | 1.24 | 3.95 | 1.38 | 1.27 | 2.91 |

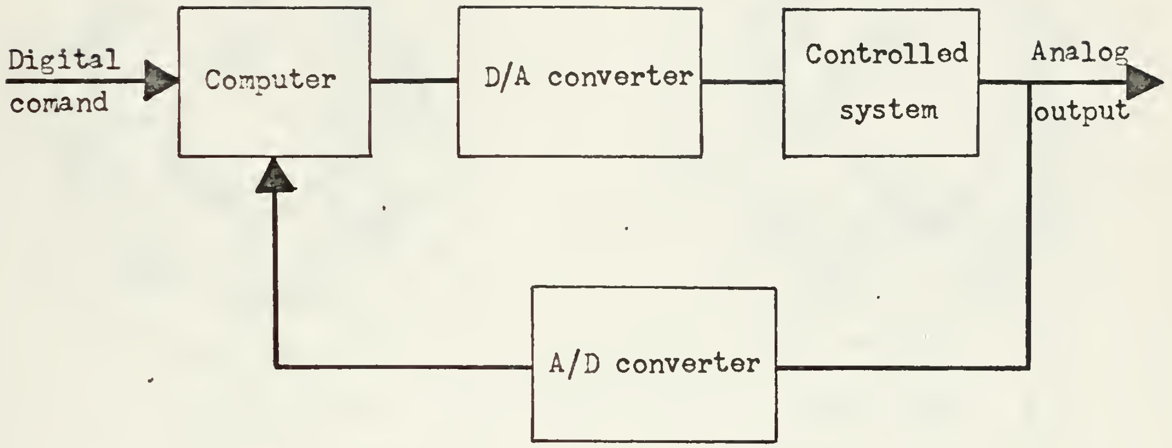



Figure 1. Block diagram for a typical digital control system.

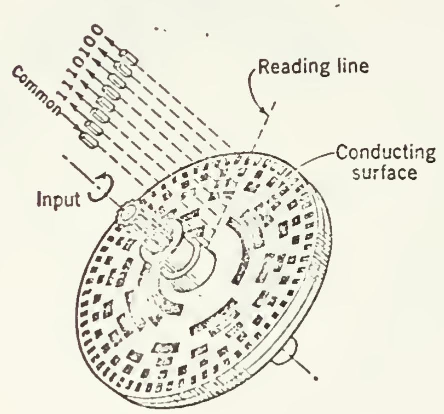



Figure 2. A typical shaft position-to-digital converter

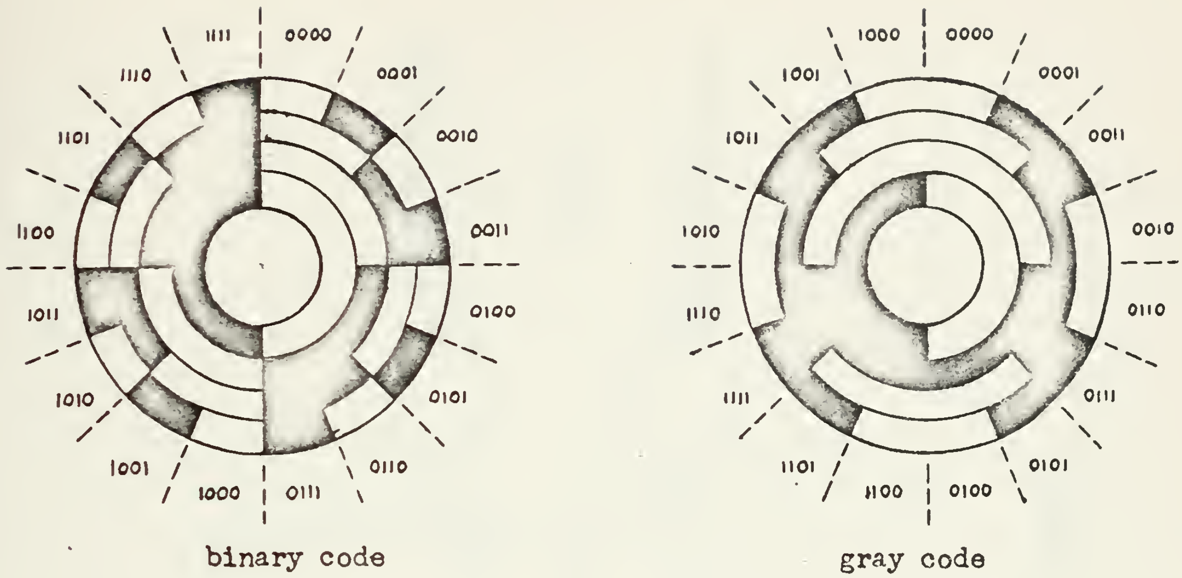



Figure 3. Two typical code disks.

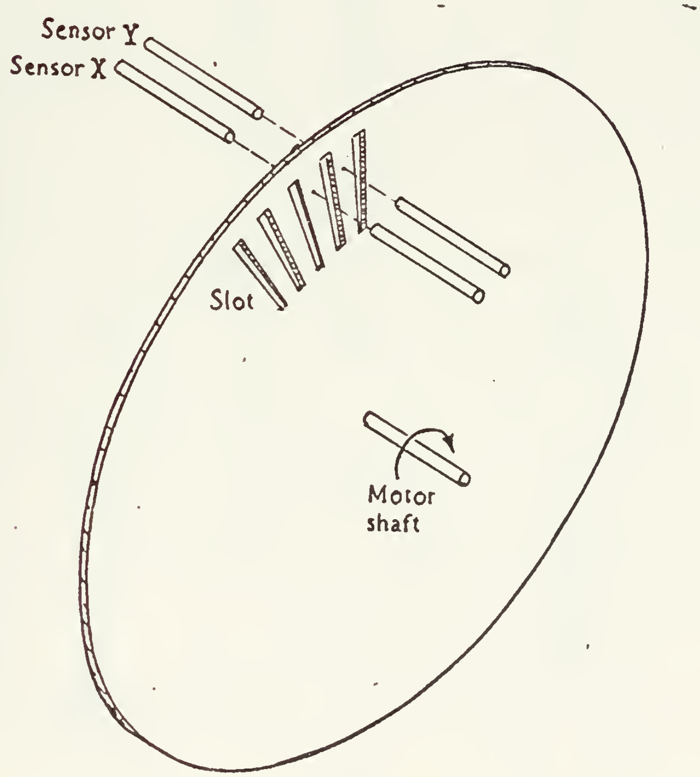



Figure 4. Code disk and photoelectric sensor units.

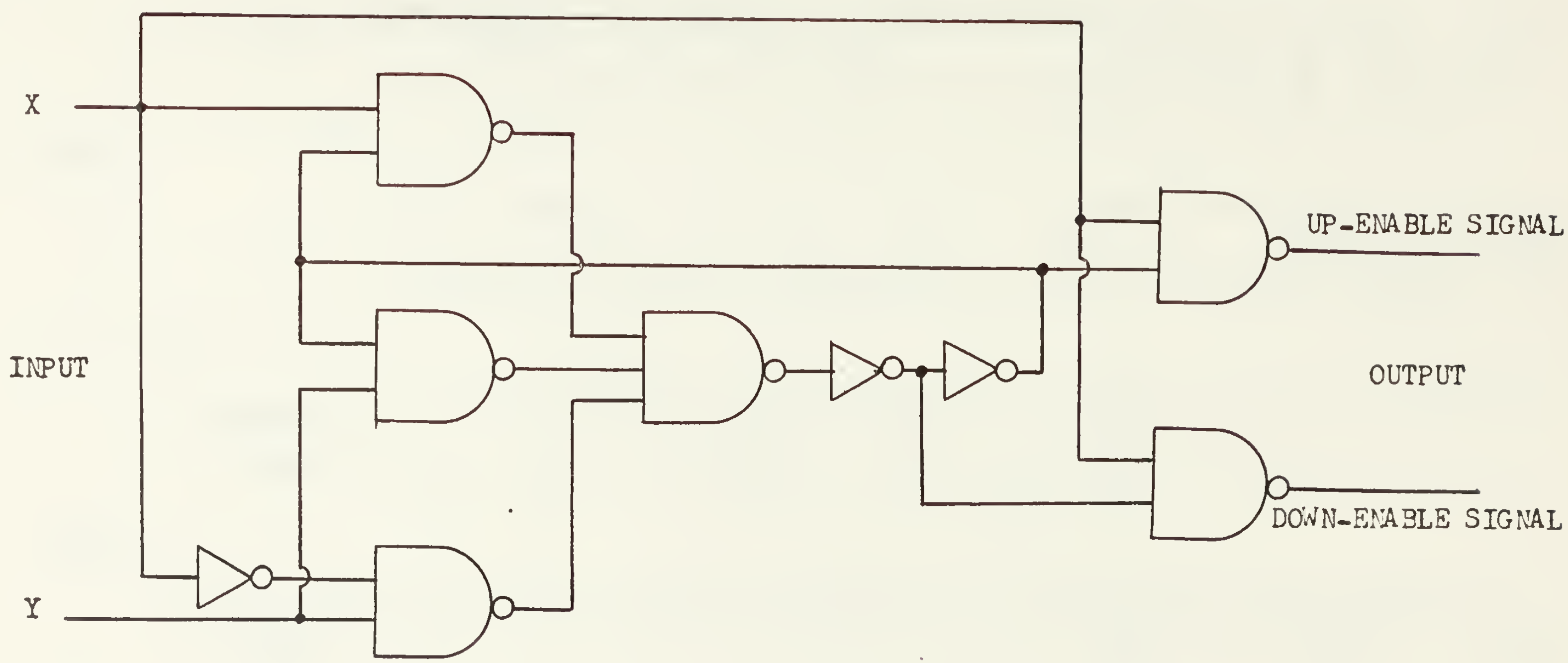



Figure 5. Direction-sensing circuit.

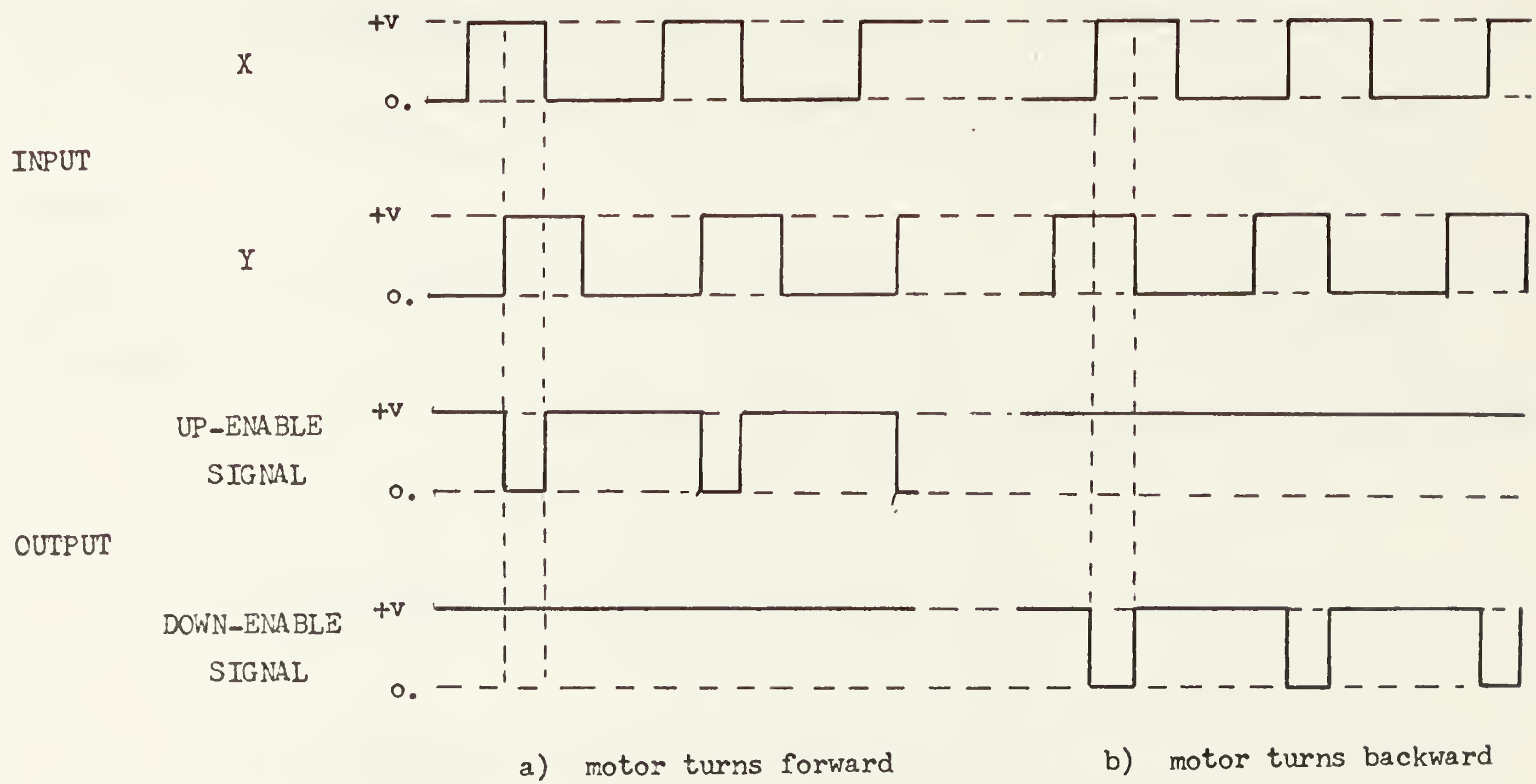



Figure 6. Input and output signals of the direction-sensing circuit

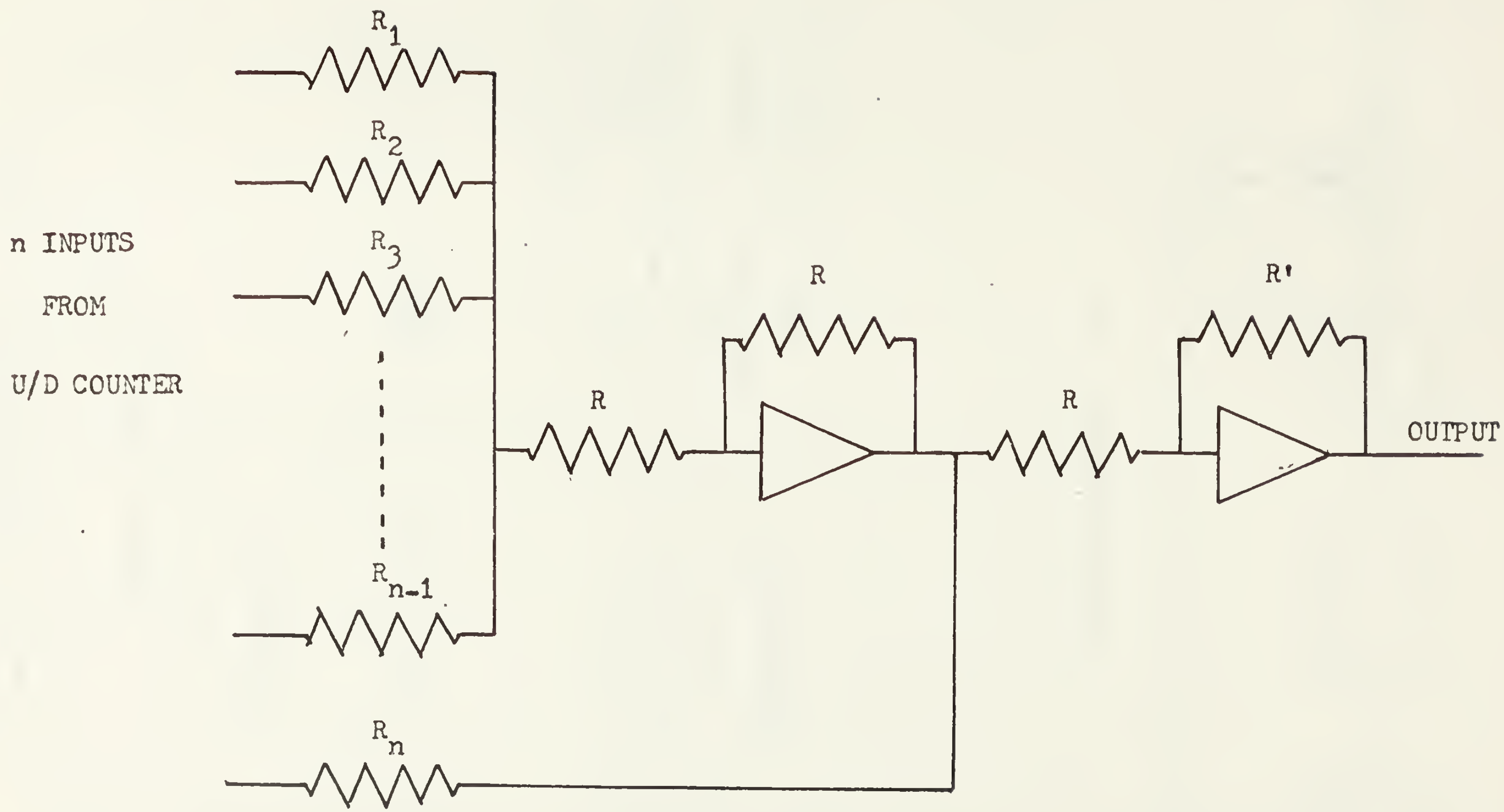
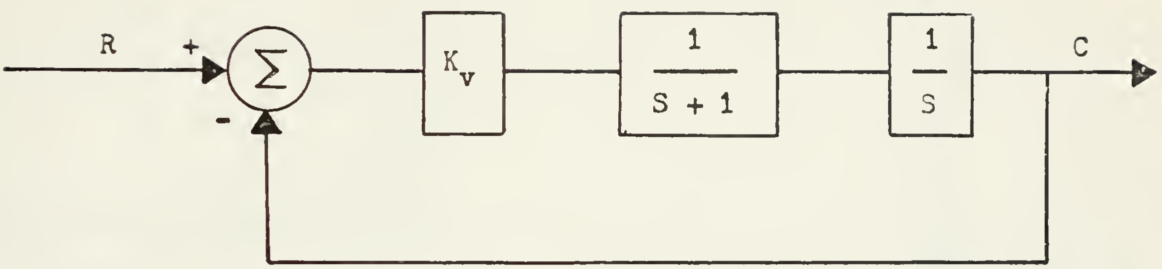
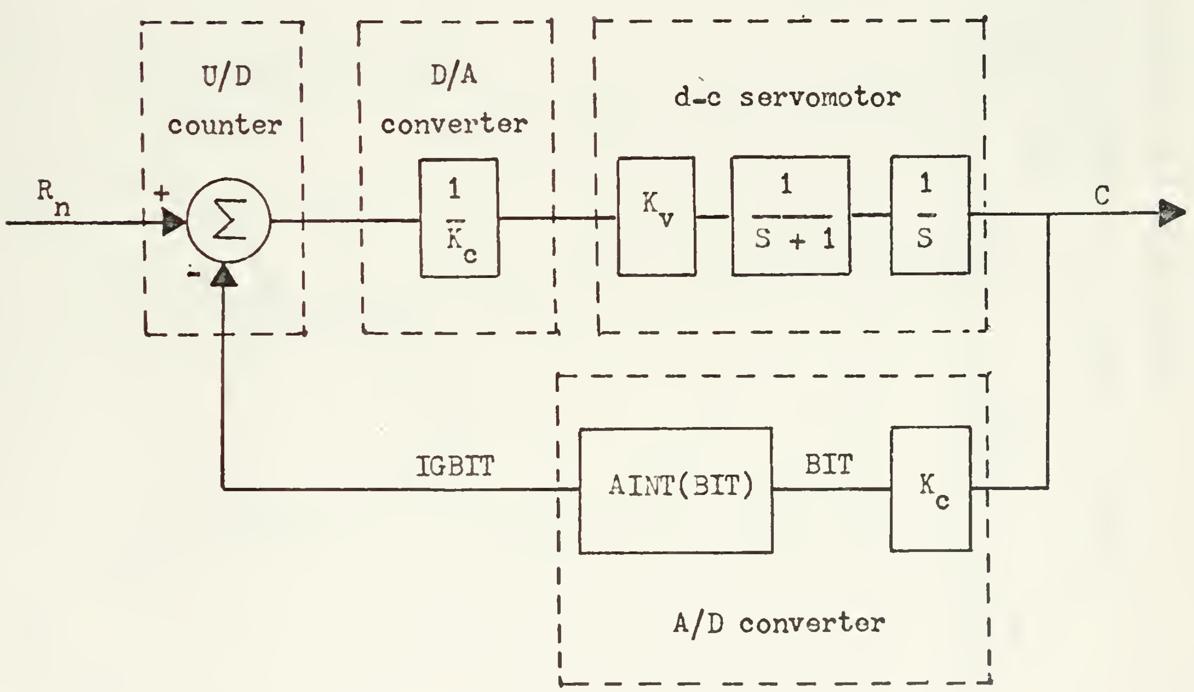



Figure 7. Digital-to-analog converter

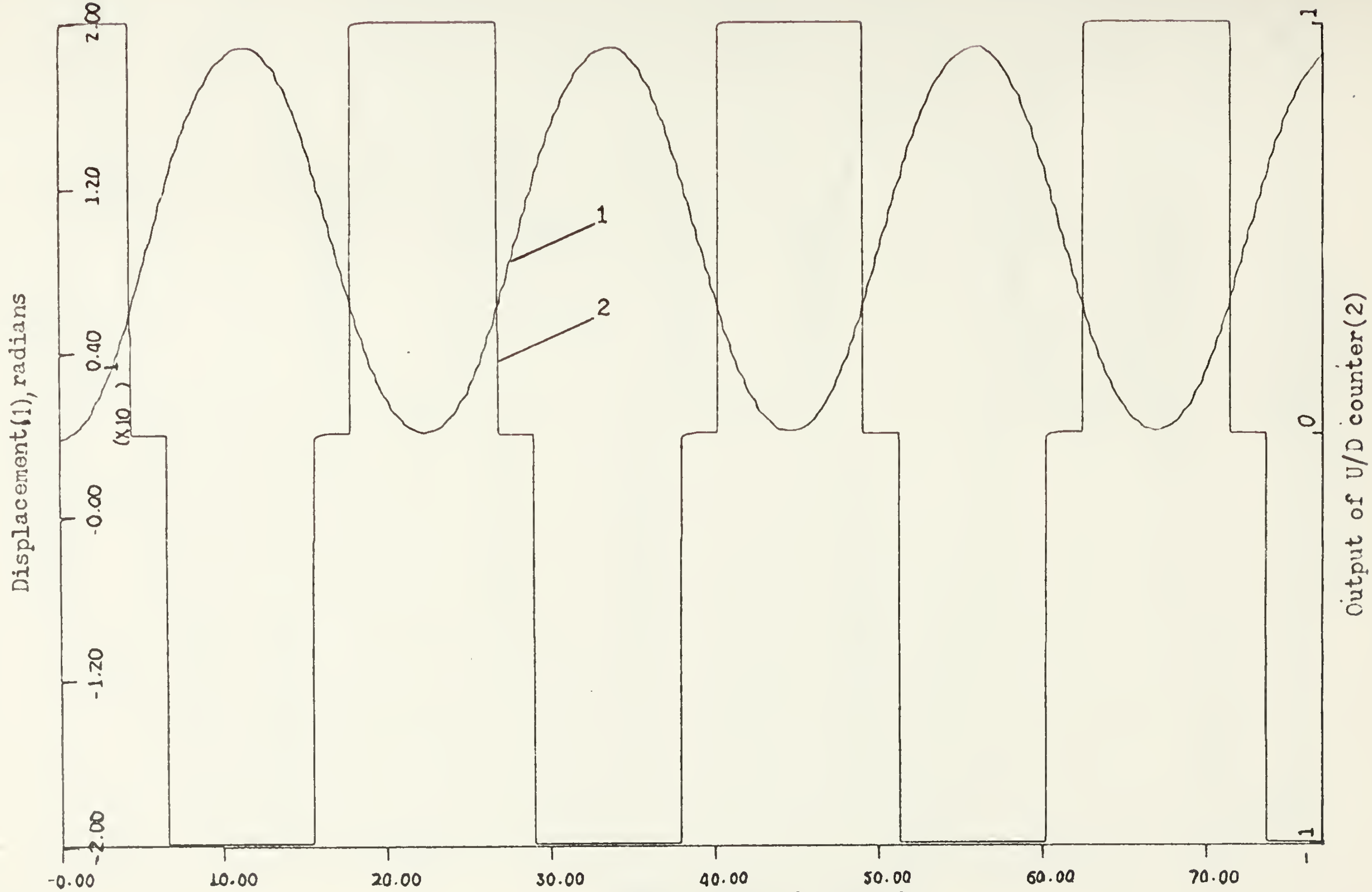



a) continuous system




b) digital system

Figure 8. Positioning servo system in single function block diagram




Time, (X10 ) $^{-9}$ second

Figure 9. Second order digital servo, $K_v=10^7$, $R_n=1$, OP=1 displacement and output of U/D counter vs time

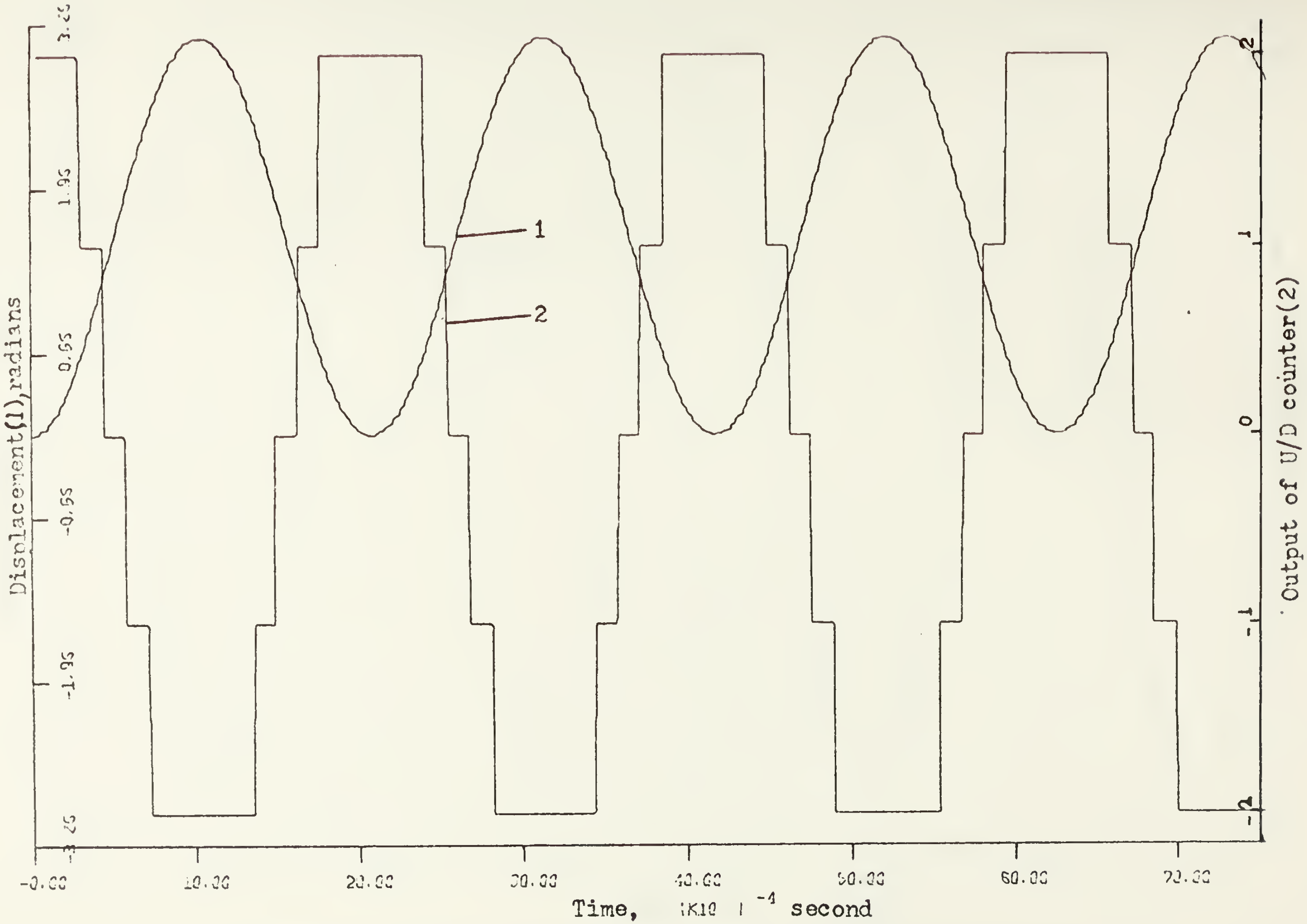



Figure 10. Second order digital servo, $K_v=10^7$, $R_n=2$, .OP=10

displacement and output of U/D counter vs time

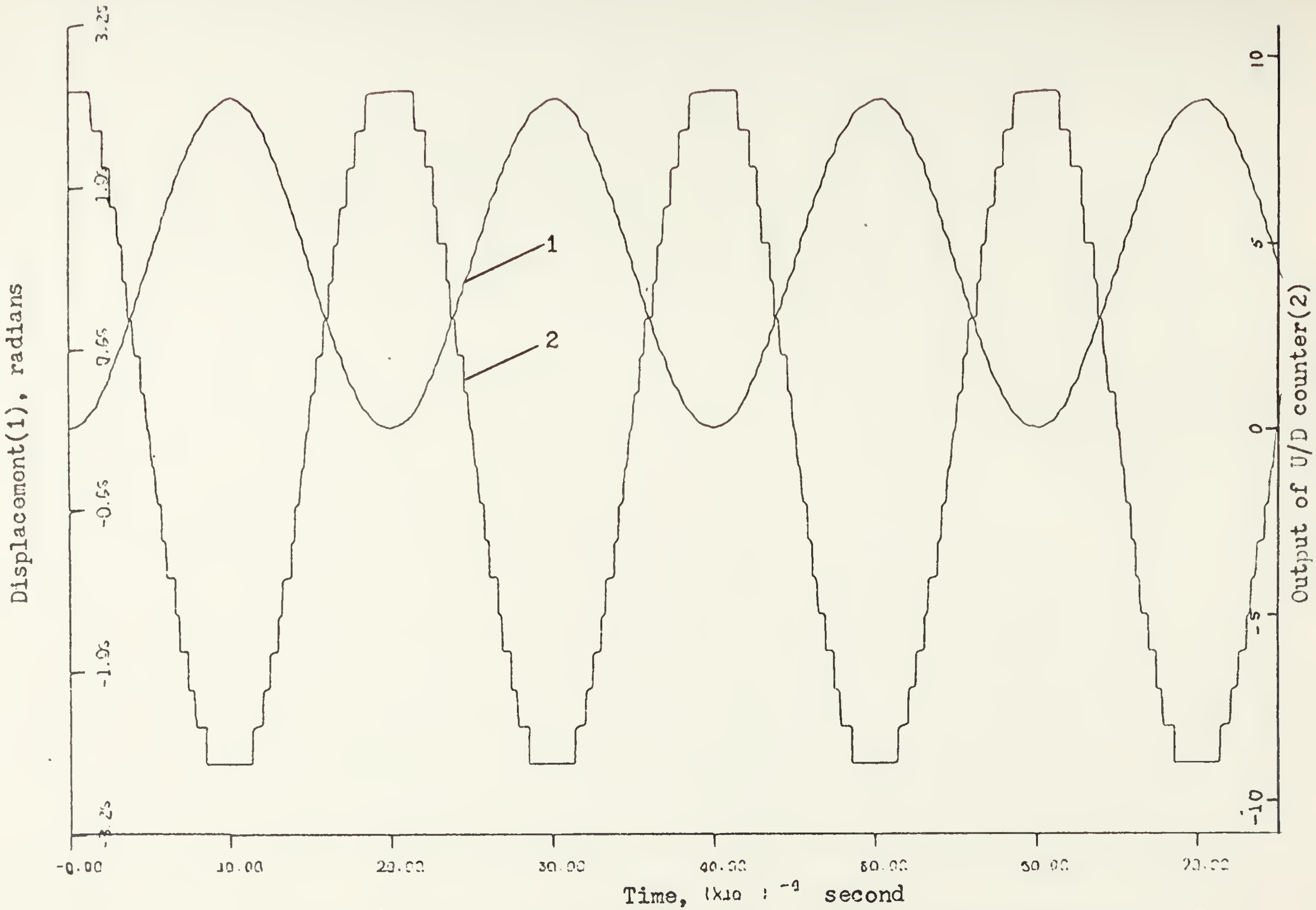



Figure 11. Second order digital servo, $K_v=10^7$, $R_n=9$, OP=45 displacement and output of U/D counter vs time

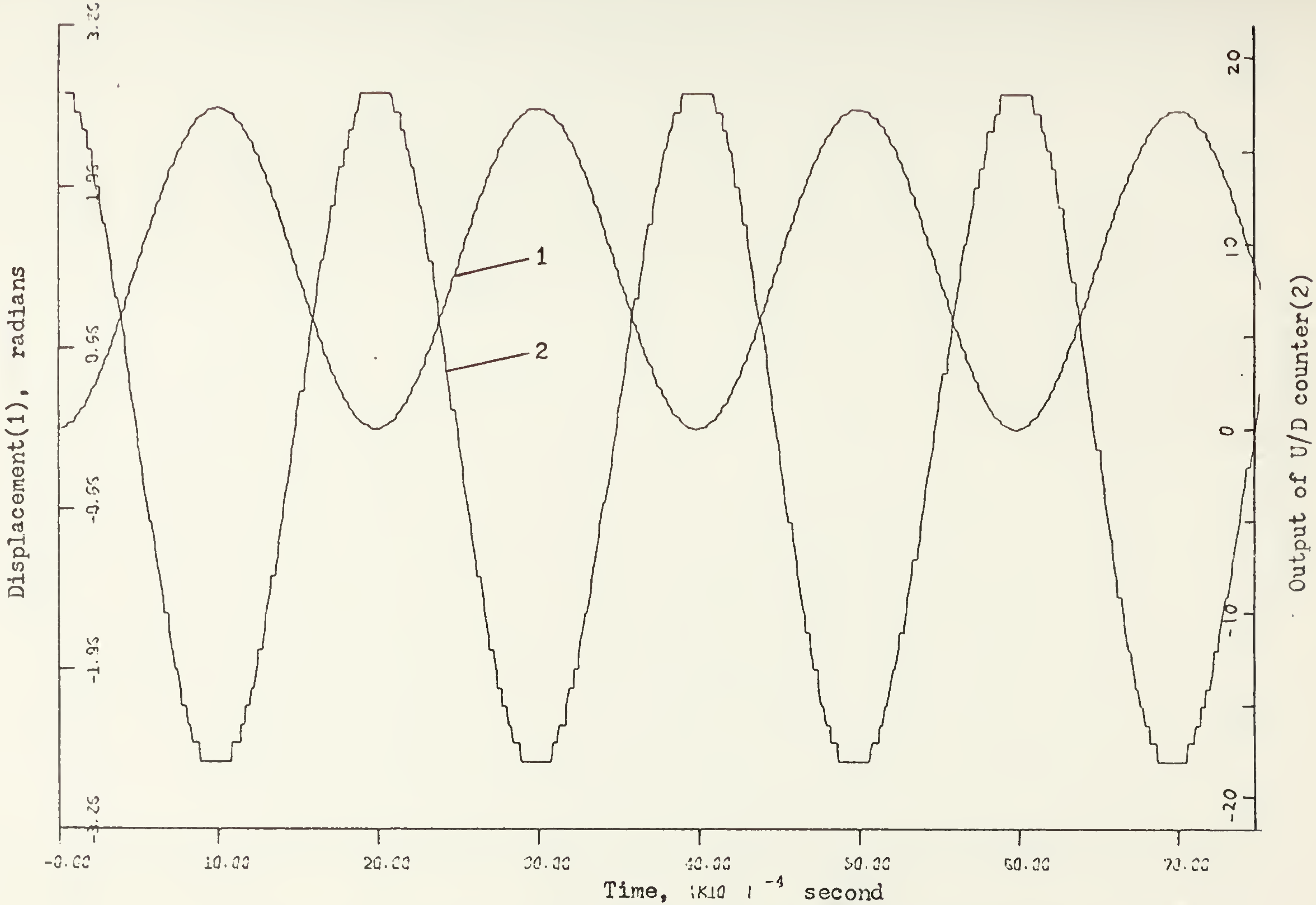



Figure 12. Second order digital servo, $K_v$=$10^7$, $R_n$=18, OP=90 displacement and output of U/D counter vs time

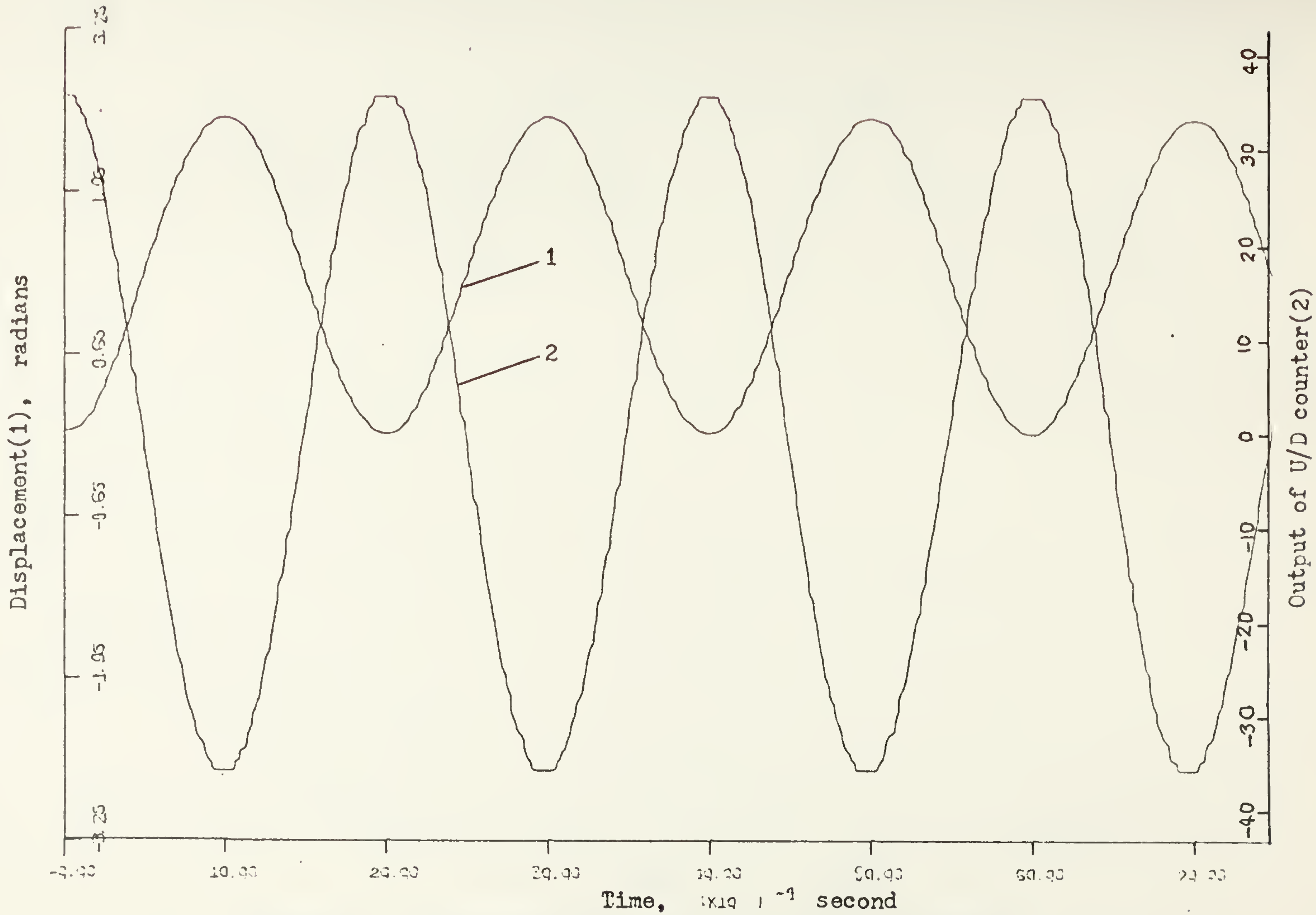



Figure 13. Second order digital servo, $K_v=10^7$, $R_n=36$, OP=180 displacement and output of U/D counter vs time

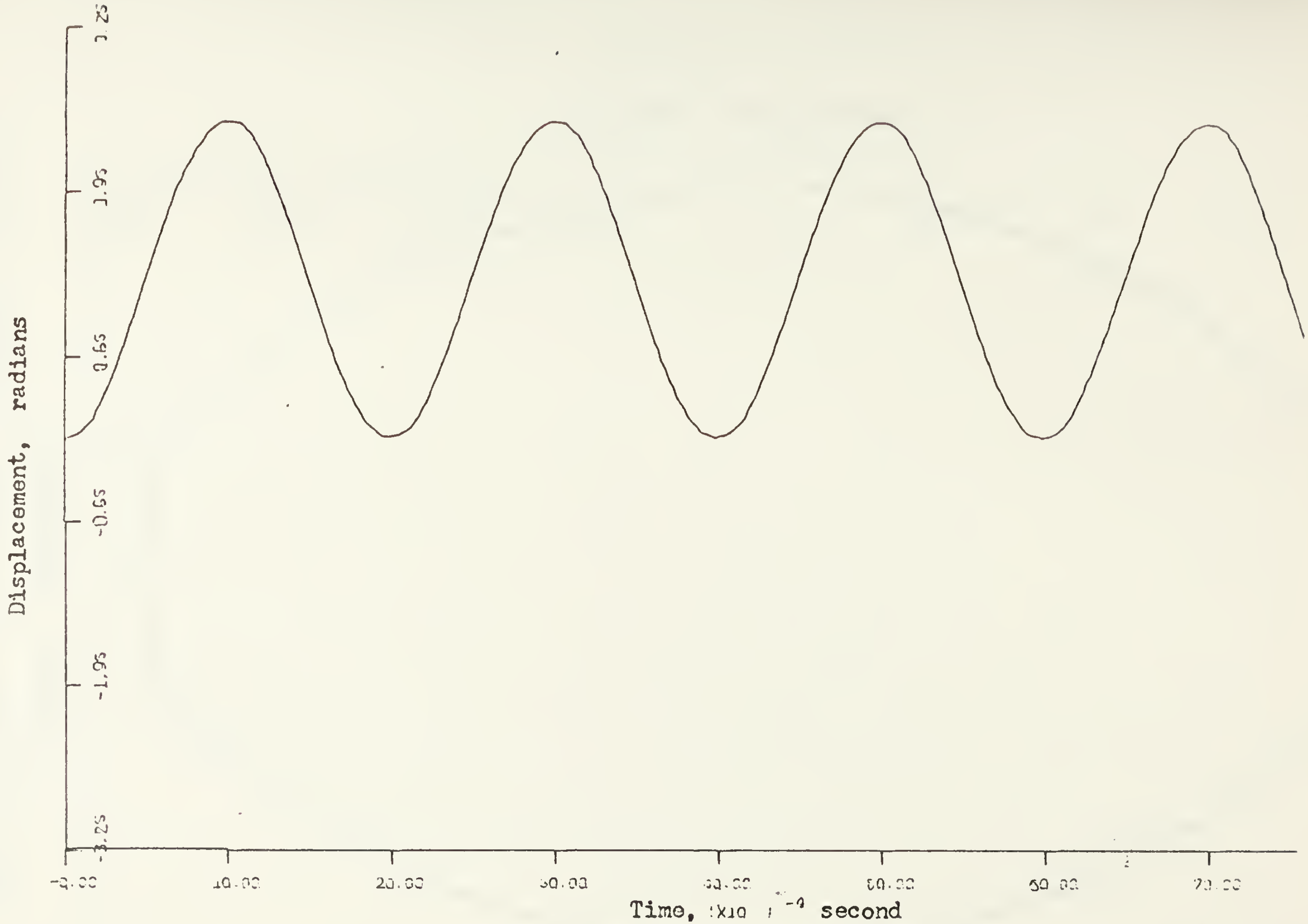



Figure 14. Second order analog servo, $K_v=10^7$, R=1.25664

displacement vs time

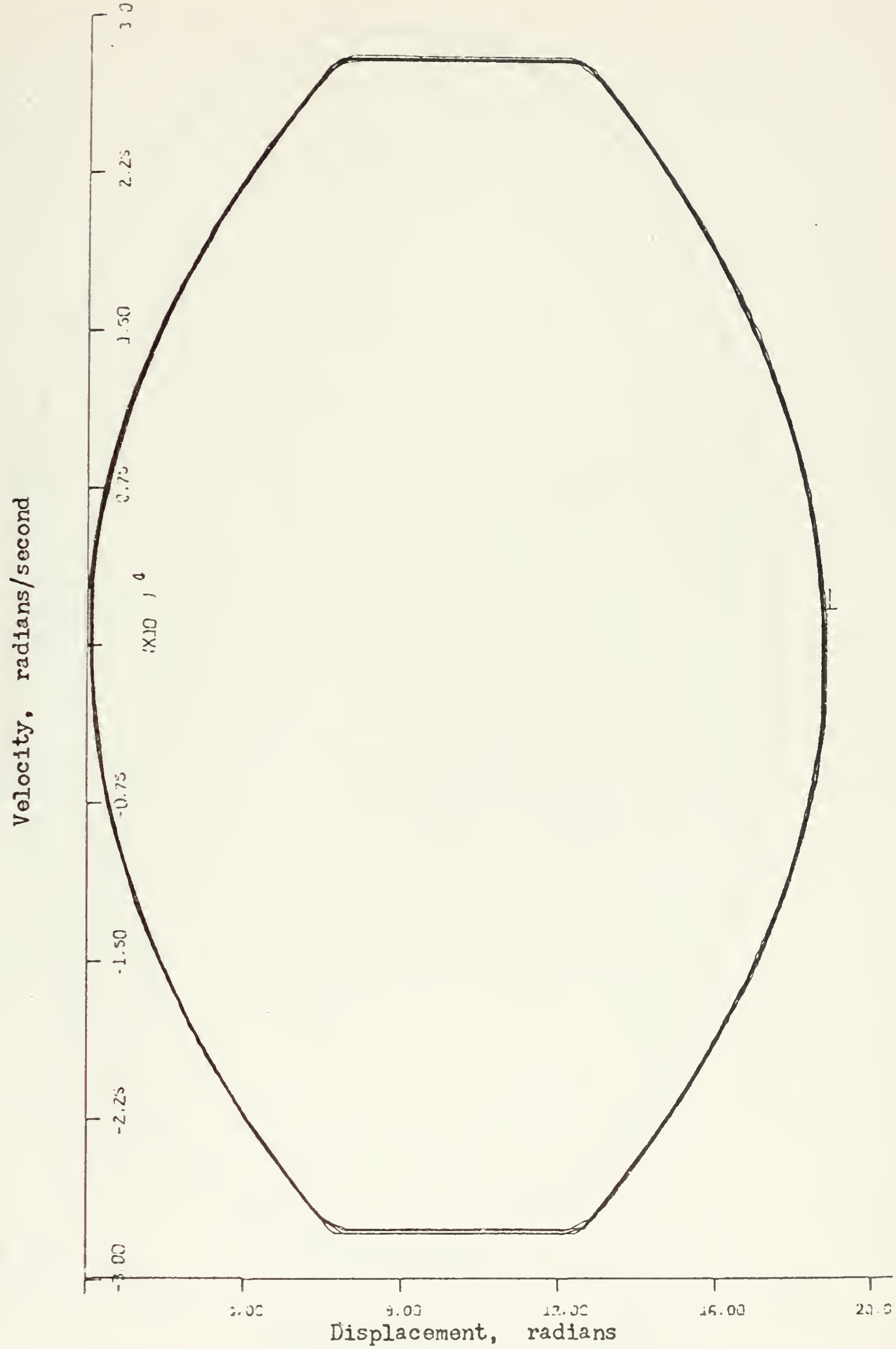



Figure 15. Second order digital servo, $K_v=10^7$, $R_n=1$, OP=1 phase plane plot


Figure 16. Second order digital servo, $K_v=10^7$, $R_n=2$, OP=10 phase plane plot


Figure 17. Second order digital servo, $K_v=10^7$, $R_n=9$, OP=45 phase plane plot

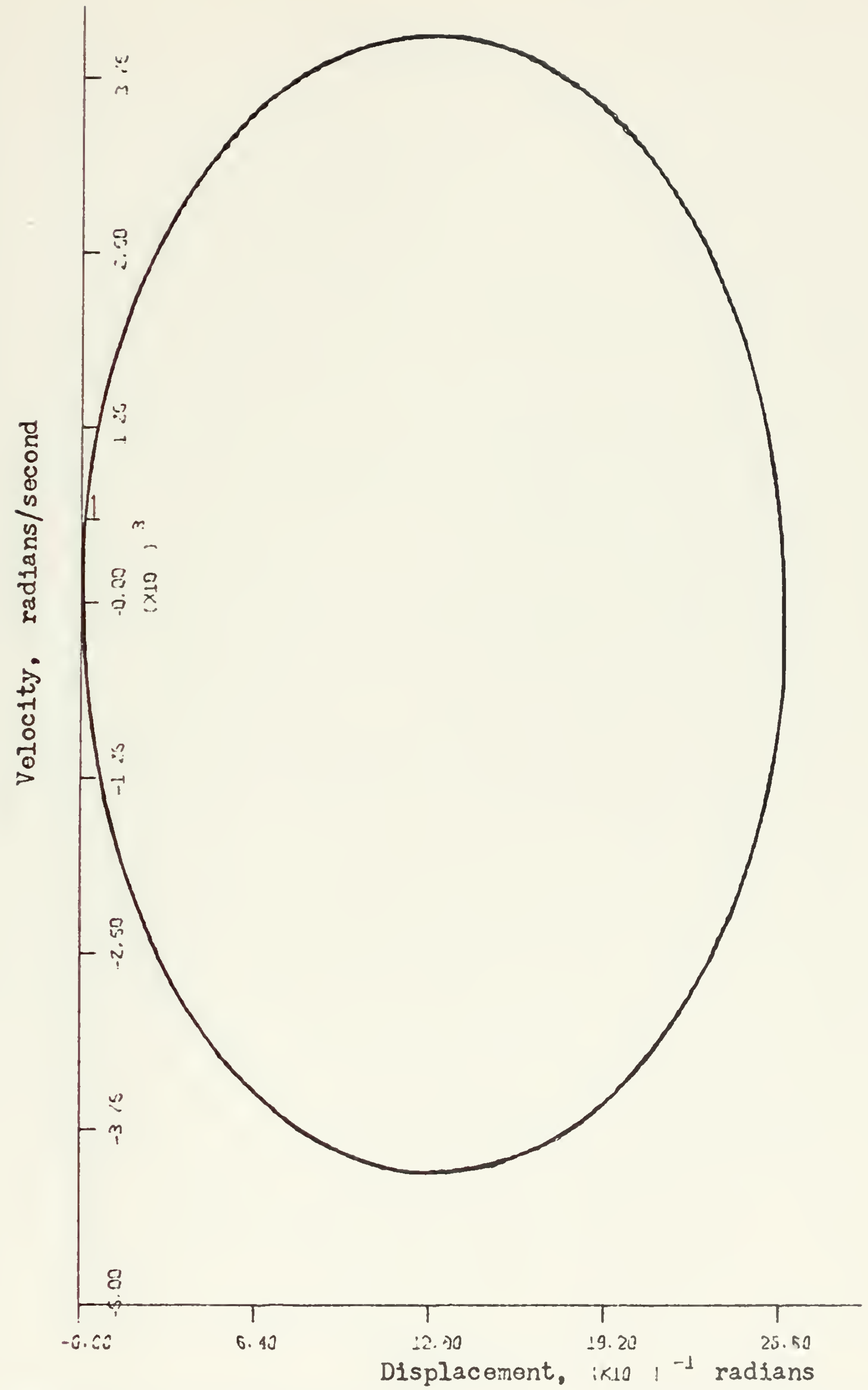



Figure 18. Second order digital servo, $K_v=10^7$, $R_n=18$, OP=90 phase plane plot

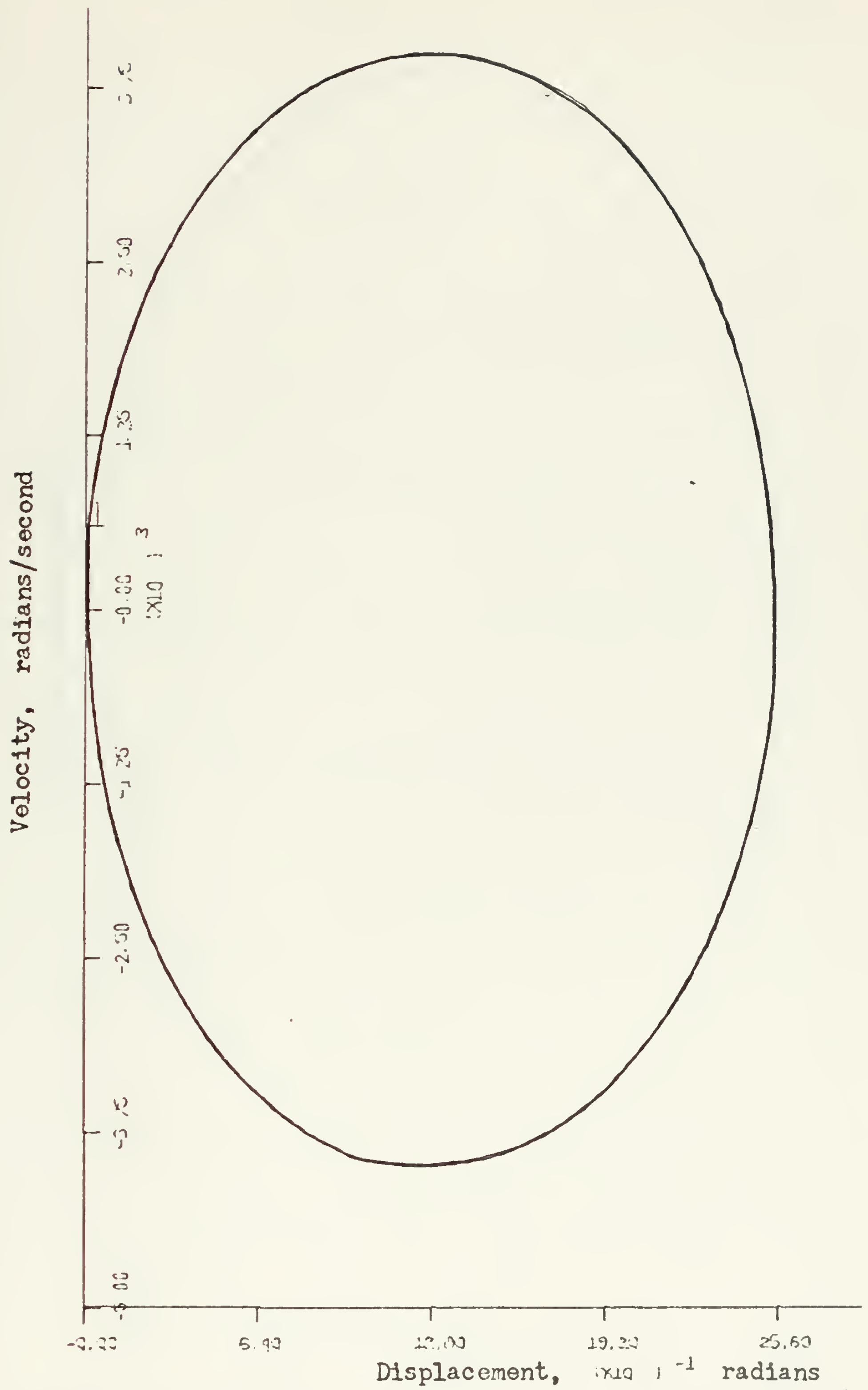



Figure 19. Second order digital servo, $K_v=10^7$, $R_n=36$, CP=180
phase plane plot

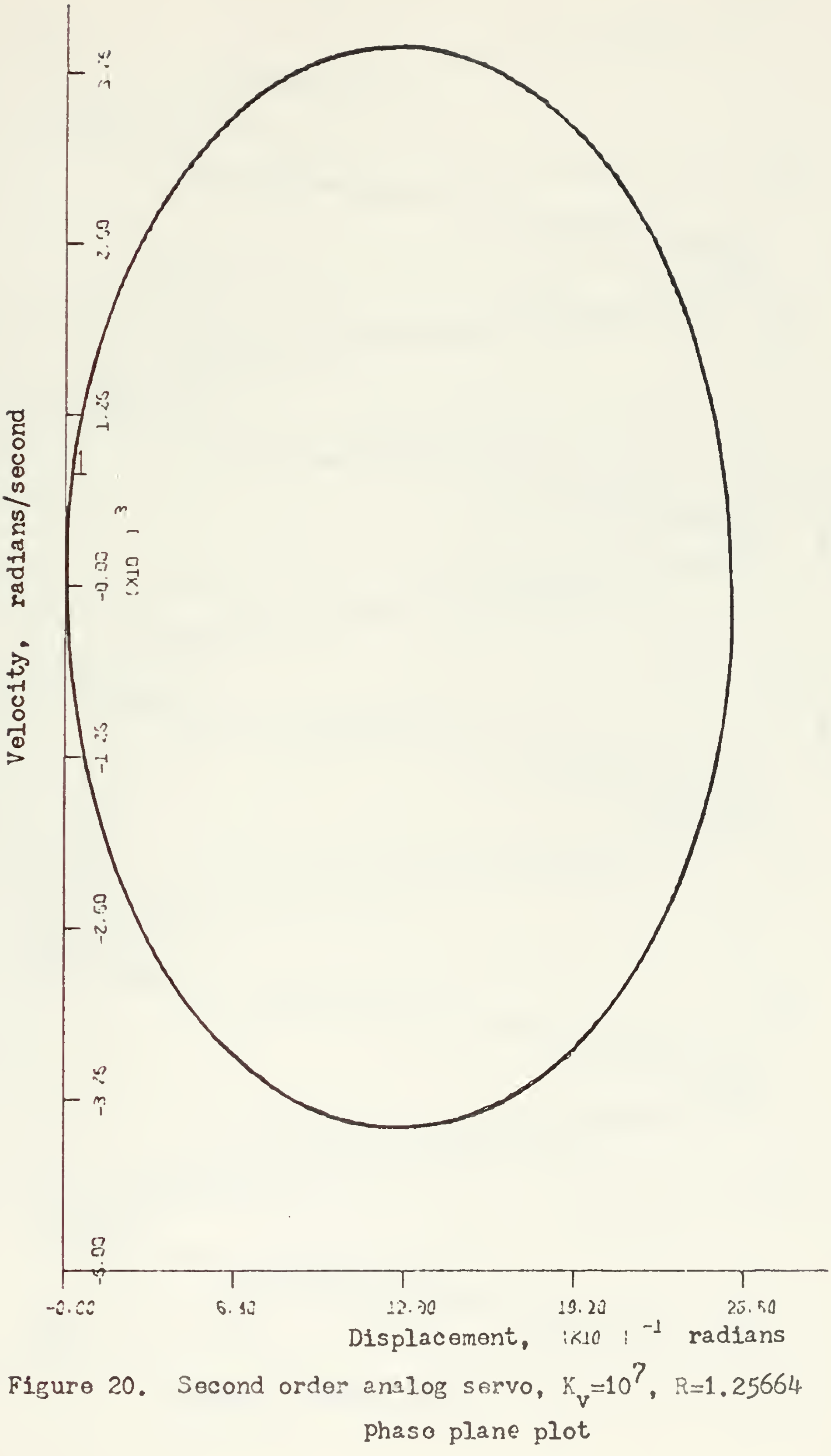



Figure 20. Second order analog servo, $K_v=10^7$, R=1.25664
phase plane plot

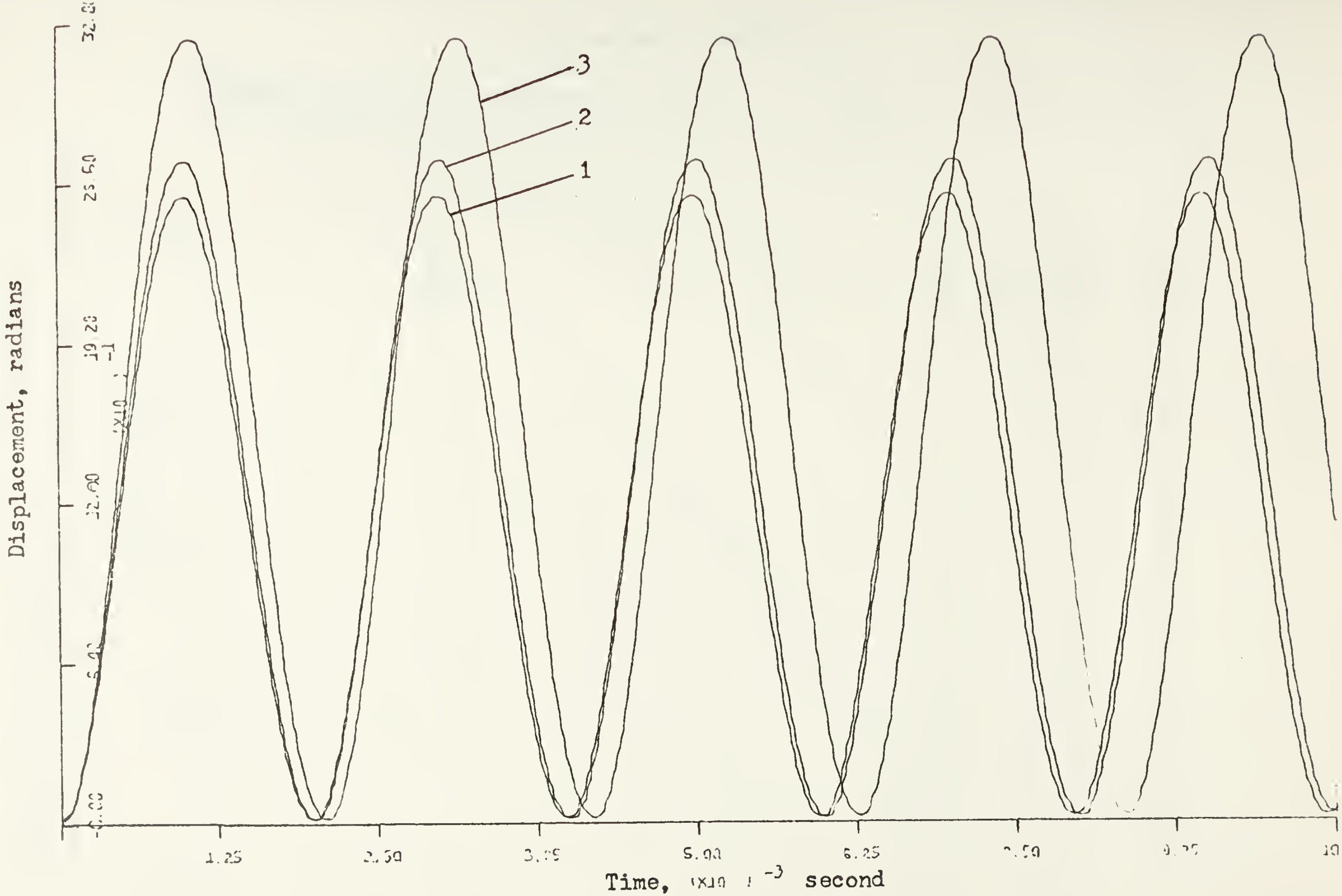



Figure 21. Step response of second order servo system, $K_v = 10^7$
analog (1), digital with OP=45 (2), and OP=10 (3)

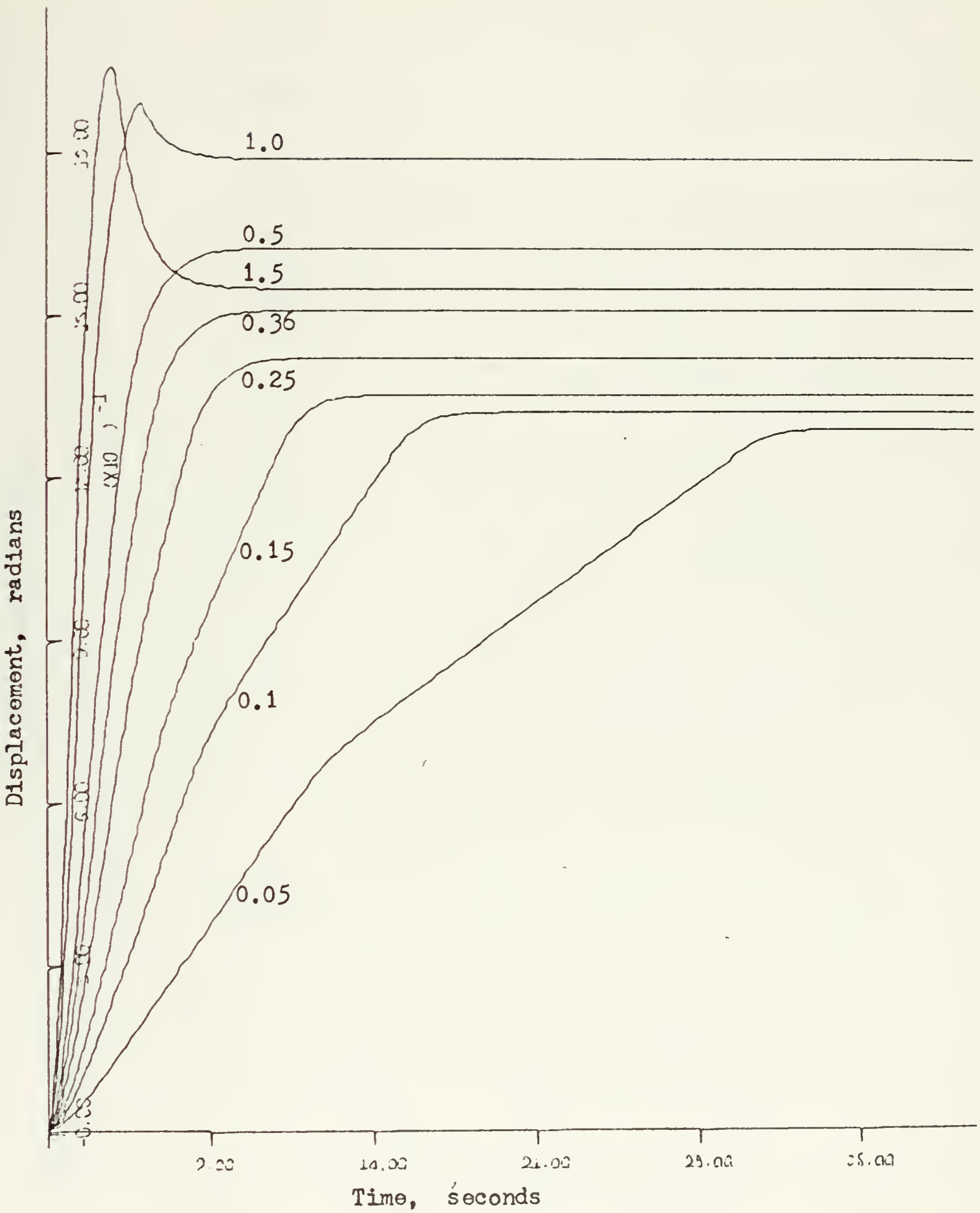
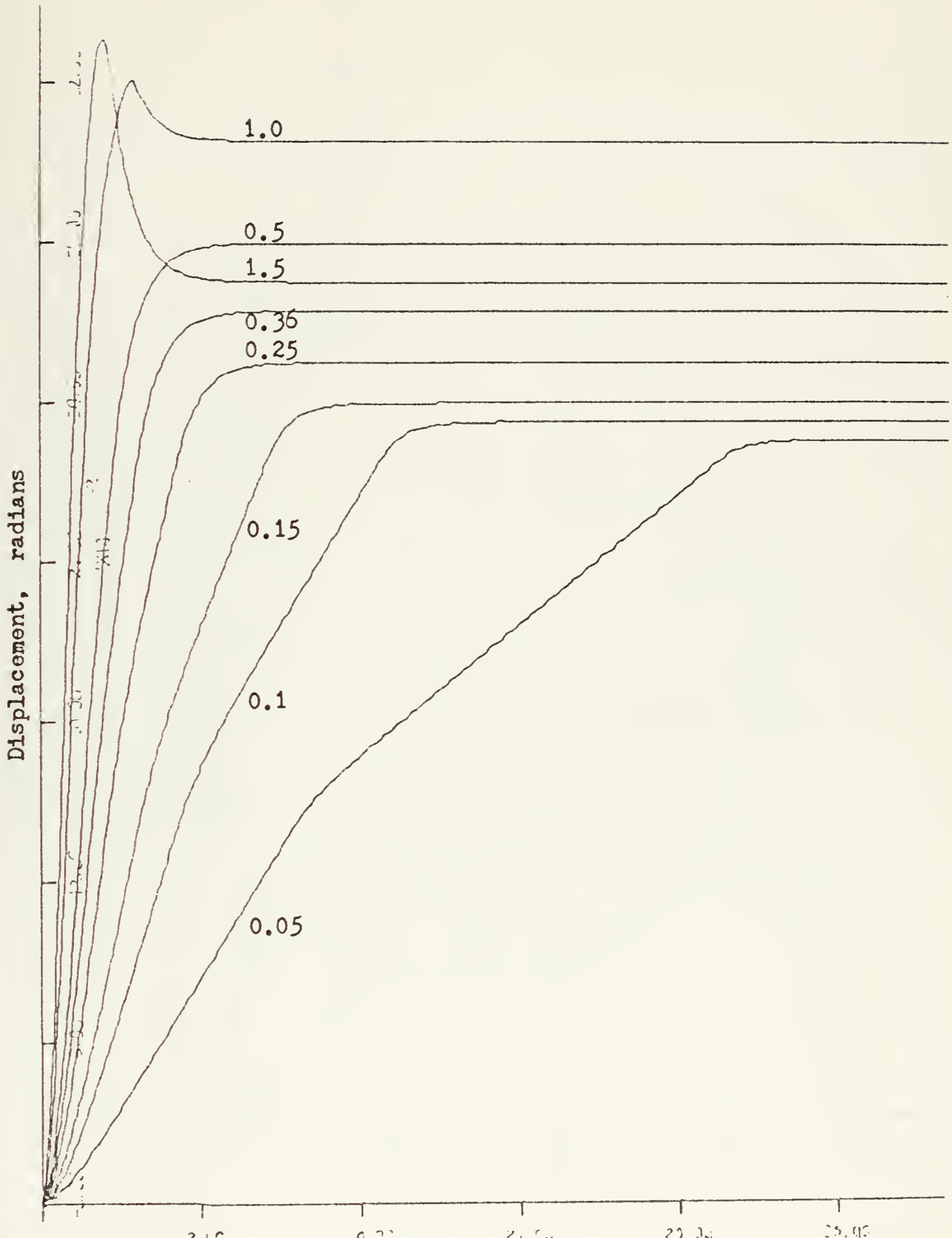



Figure 22. Step response of second order digital servo $K_v$=1.5, 1.0, 0.5, 0.36, 0.25, 0.1, and 0.05, OP=10, and $R_n$=2




·Figure 23. Step response of second order digital servo $K_v$=1.5, 1.0, 0.5, 0.36, 0.25, 0.1, and 0.05, OP=45, and $R_n$=2

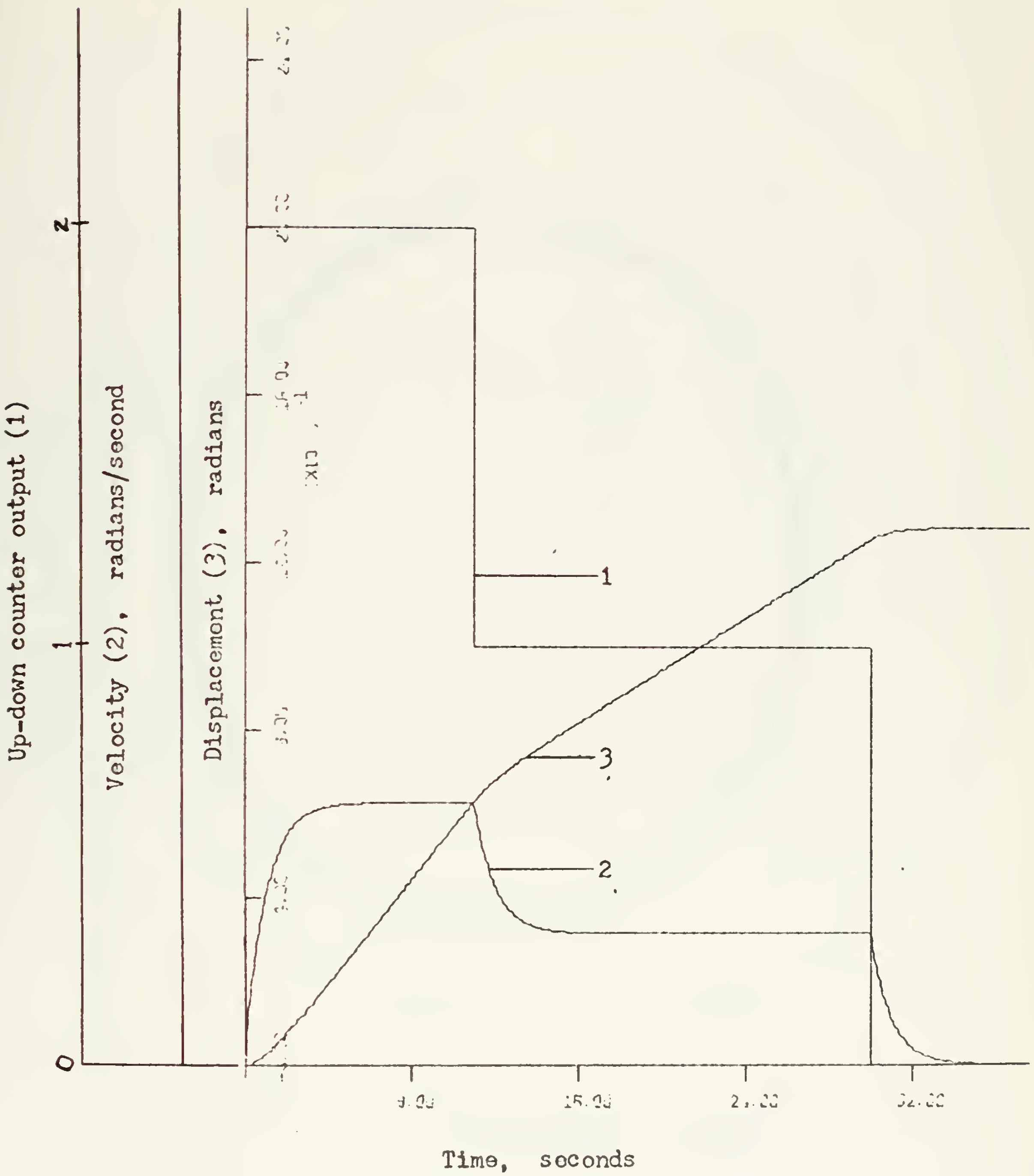



Figure 24. Second order digital servo, $K_v$=0.05, OP=10, and $R_n$=2
Up-down counter output, velocity, and displacement vs time

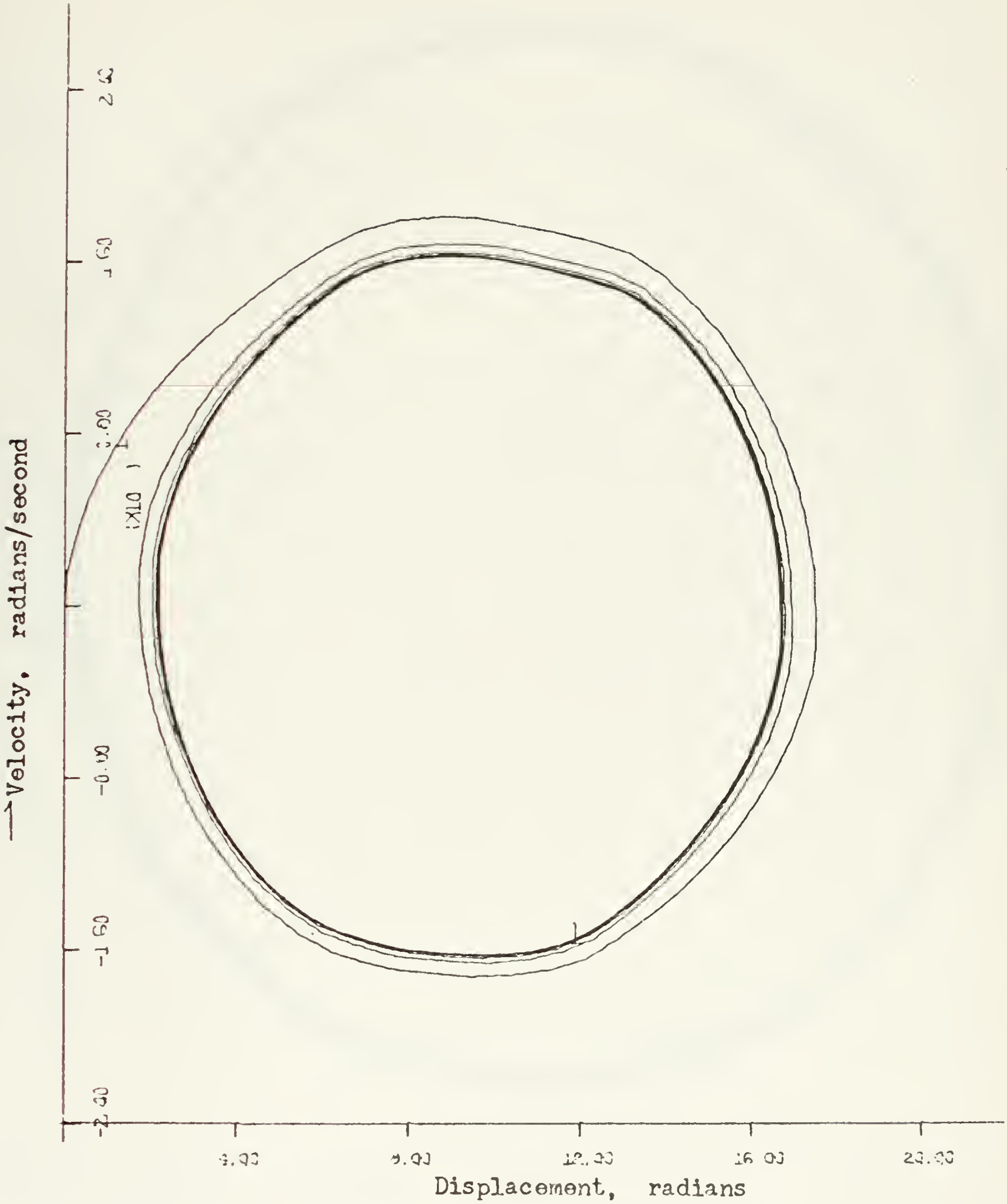



Figure 25. Third order digital servo, $K_v$=6, $R_n$=1, OP=1 phase plane plot

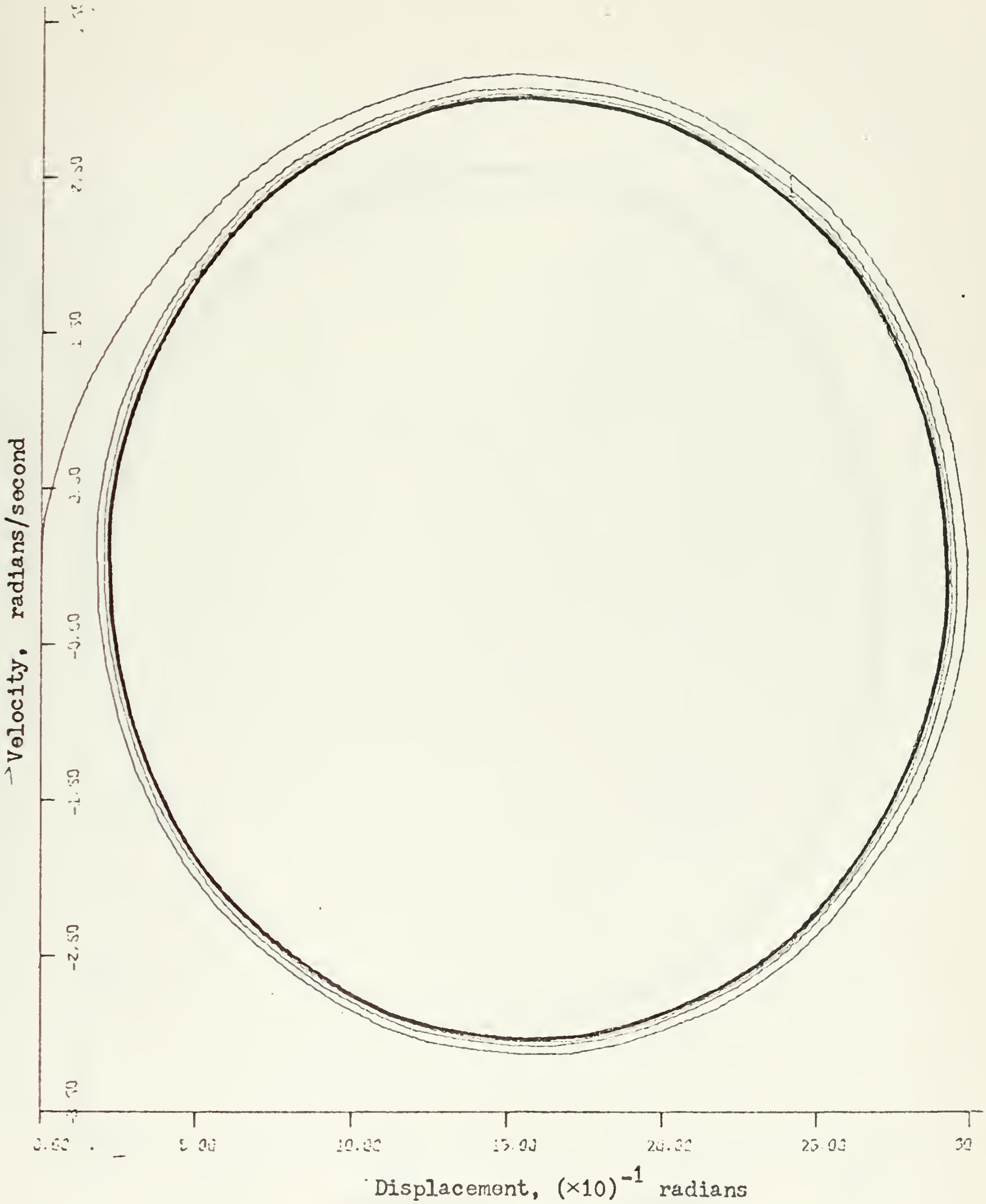



Figure 26. Third order digital servo, K =6, R =2, OP=10
phase plane plot

Figure 27. Third order digital servo, $K_v=6$, $R_n=9$, OP=45 phase plane plot


Figure 28. Third order digital servo, $K_v$=6, $R_n$=18, OP=90 phase plane plot


Figure 29. Third order digital servo, $K_v$=6, $R_n$=36, OP=180 phase plane plot

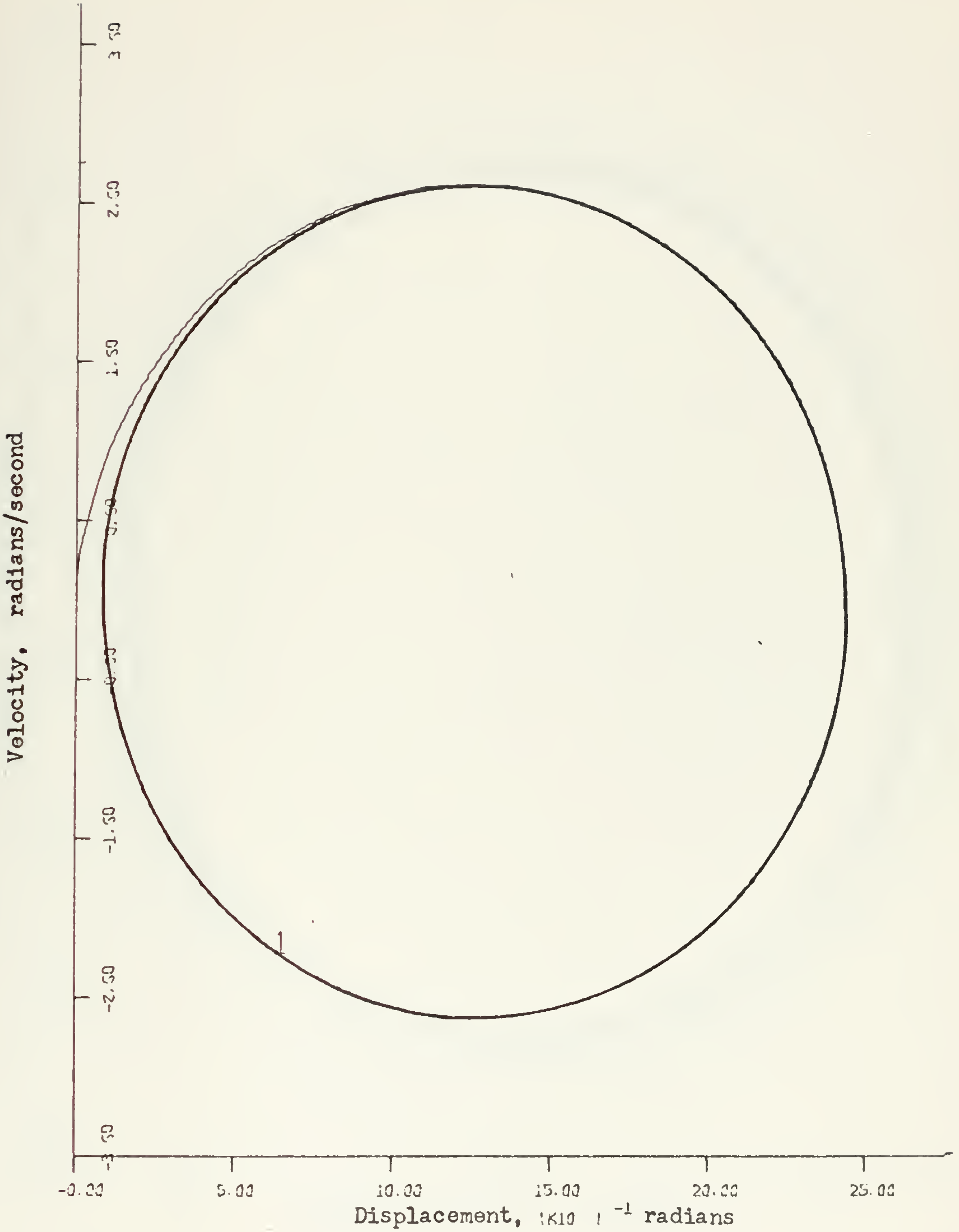



Figure 30. Third order analog servo, $K_v$=6, R=1.25664
phase plane plot

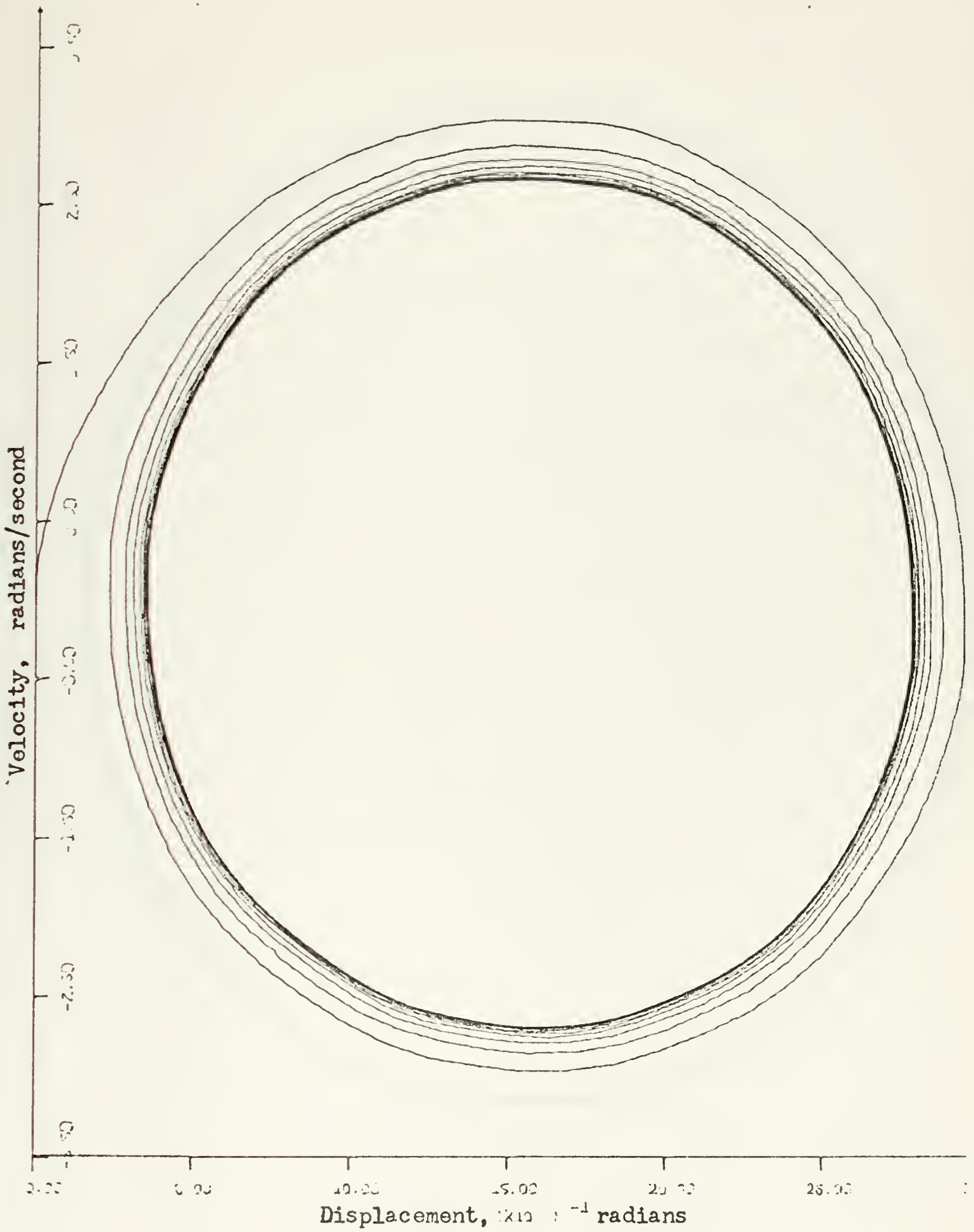



Figure 31. Third order digital servo, $K_v$=5.7, $R_n$=2, OP=10
phase plane plot

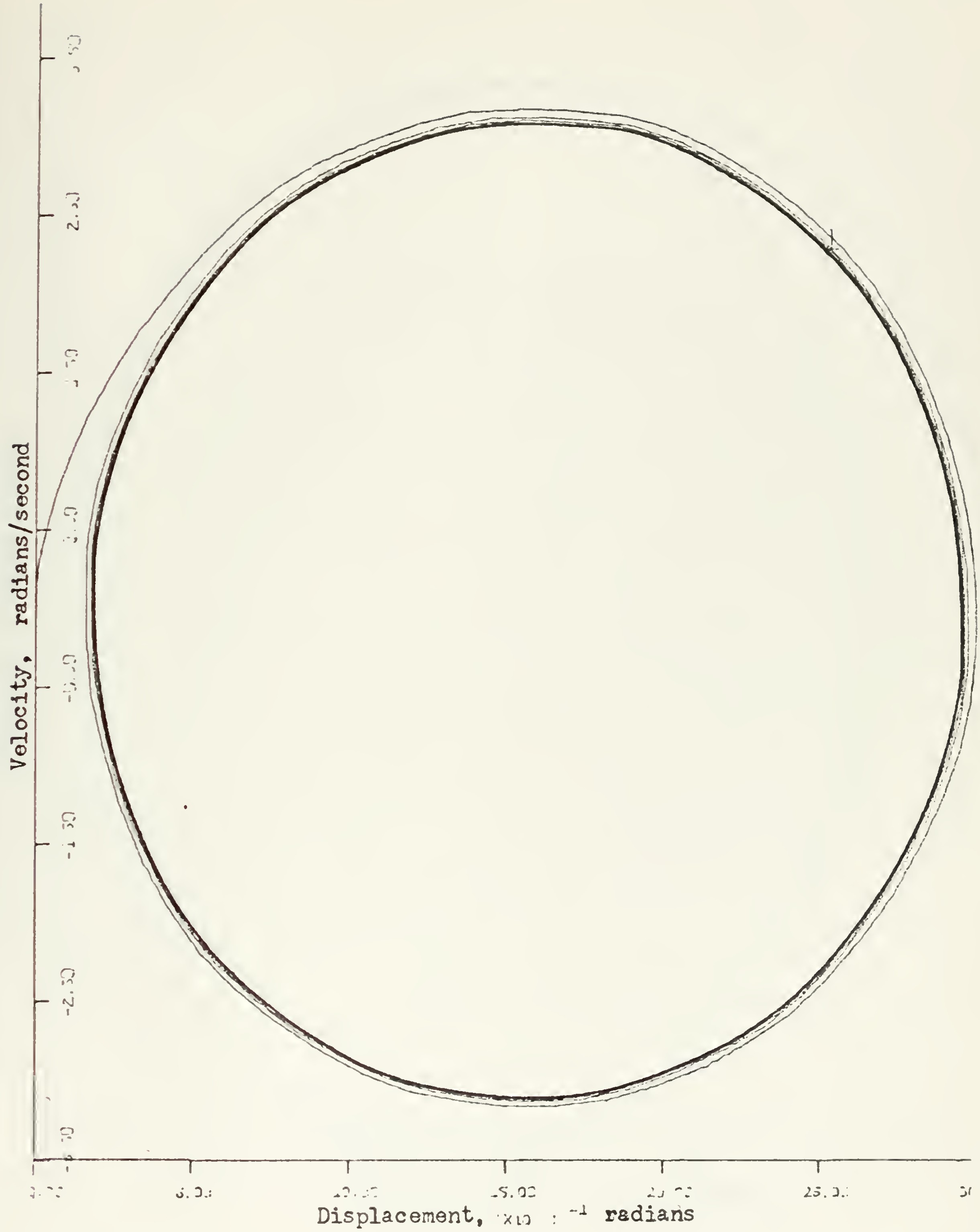


Figure 32. Third order digital servo, $K_v$=6.1, $R_n$=2, OP=10
phase plane plot

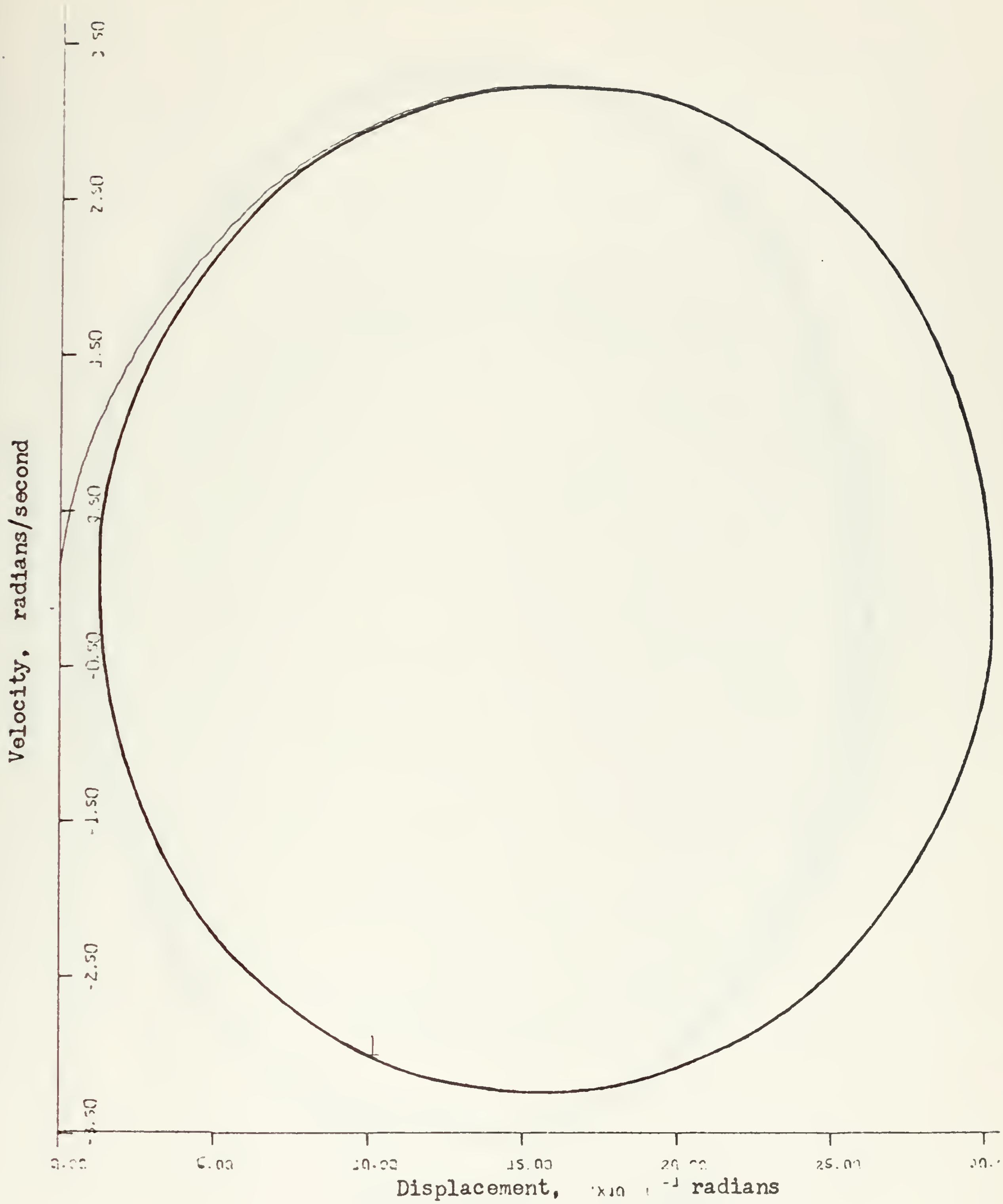



Figure 33. Third order digital servo, $K_v$=6.25, $R_n$=2, OP=10 phase plane plot

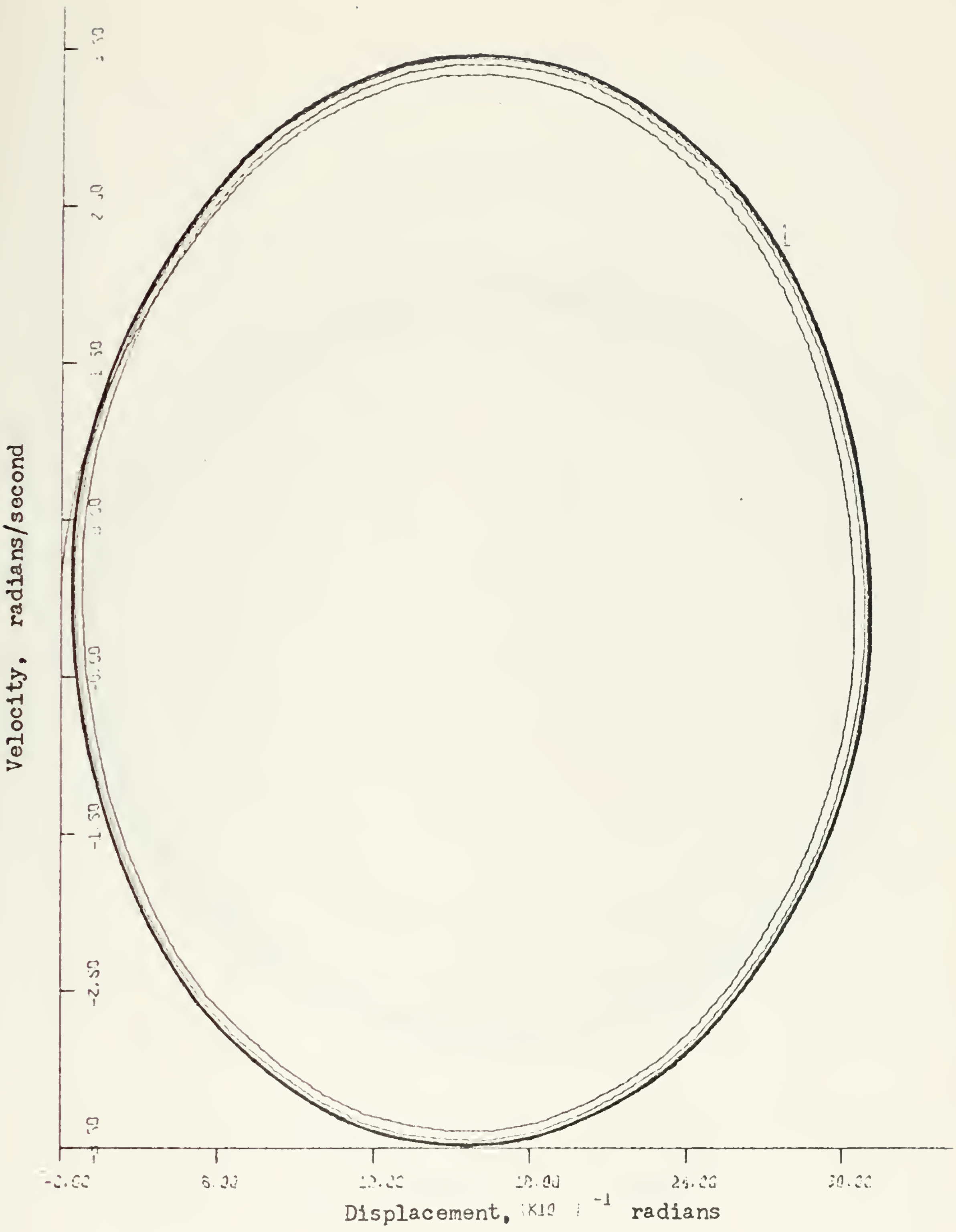



Figure 34. Third order digital servo, $K_v$=6.5, $R_n$=2, OP=10 phase plane plot

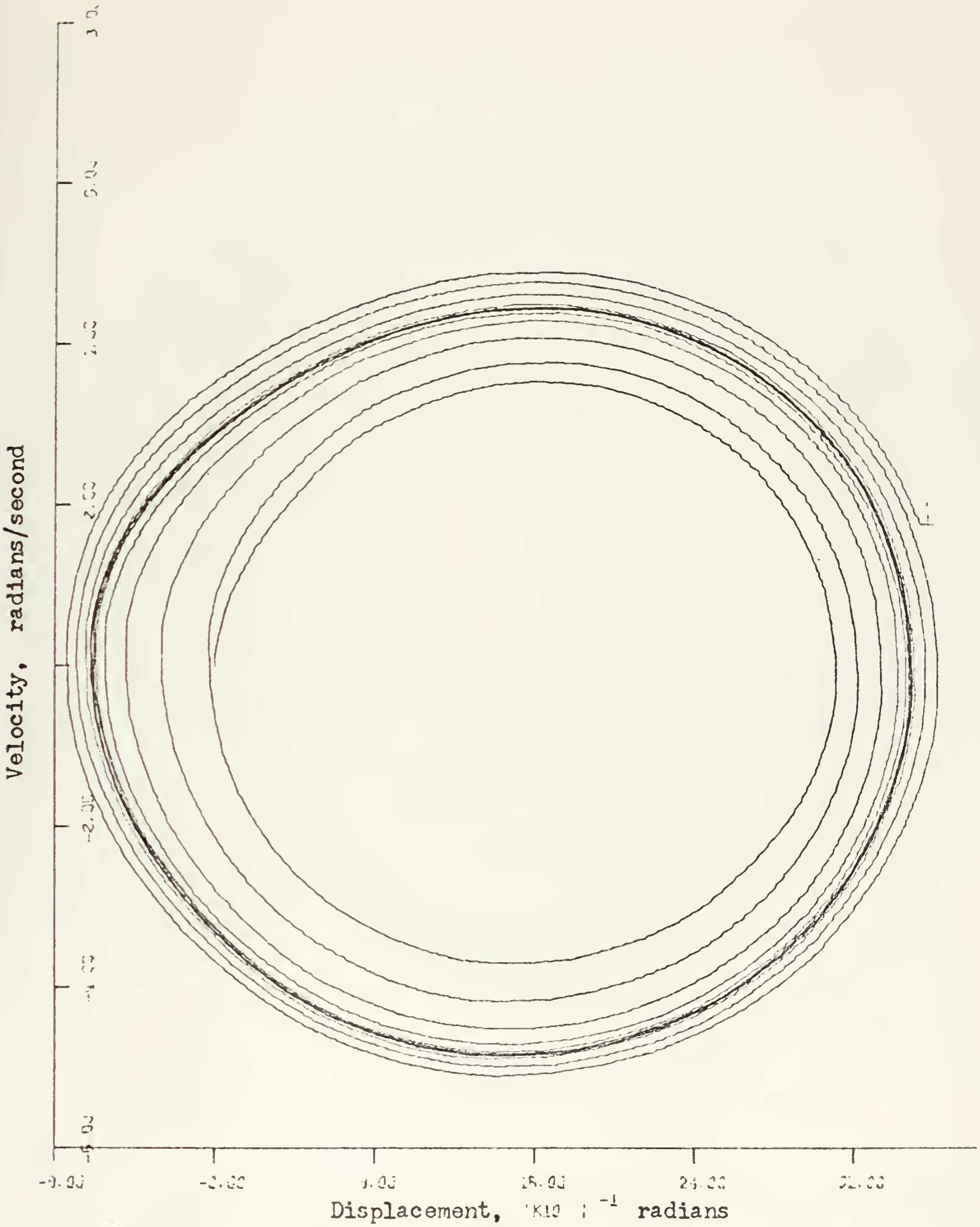



Figure 35. Third order digital servo, $K_v$=7.1, $R_n$=2, OP=10 phase plane plot

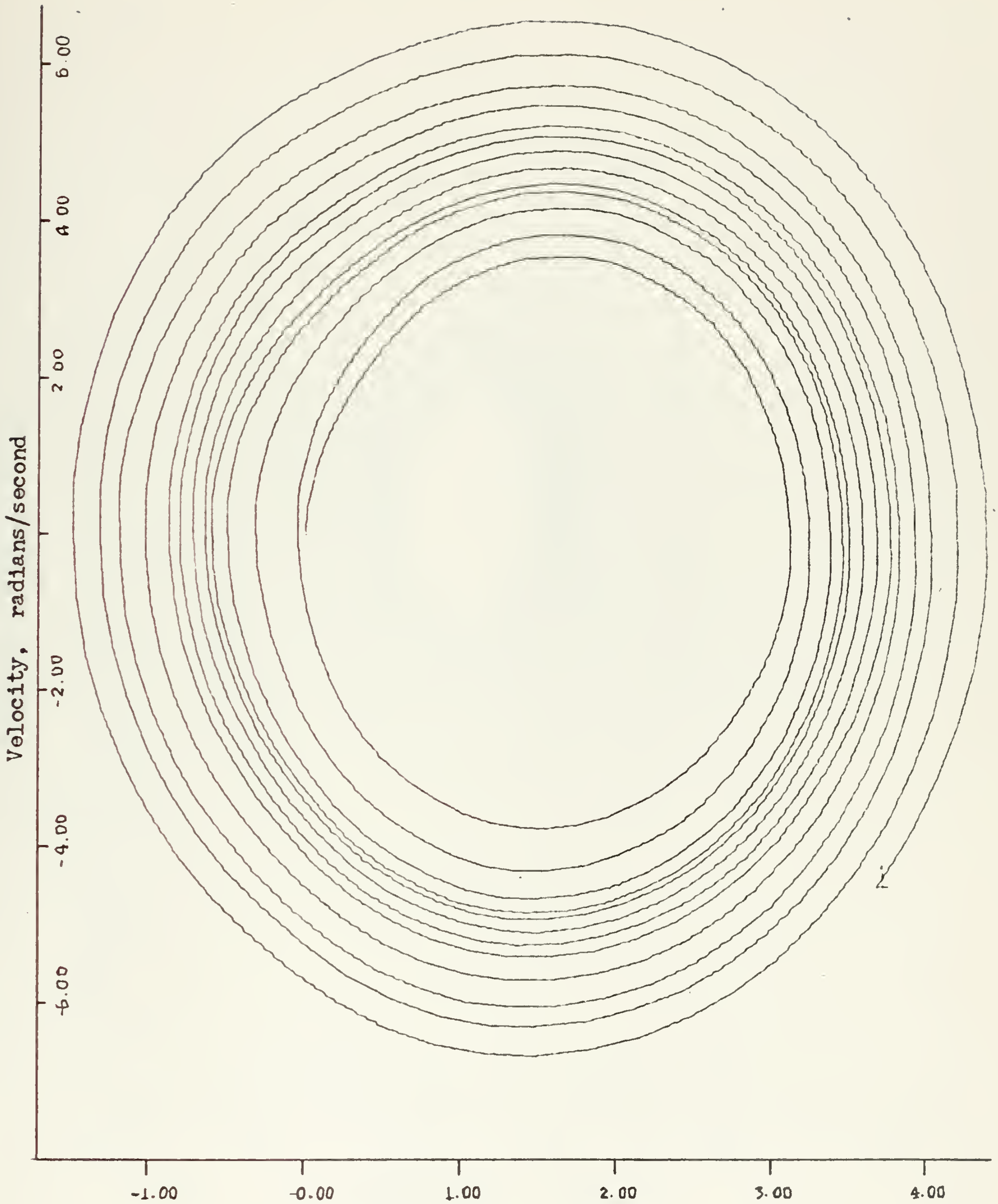



Figure 36. Third order digital servo, $K_v$=7.2, $R_n$=2, OP=10
phase plane plot

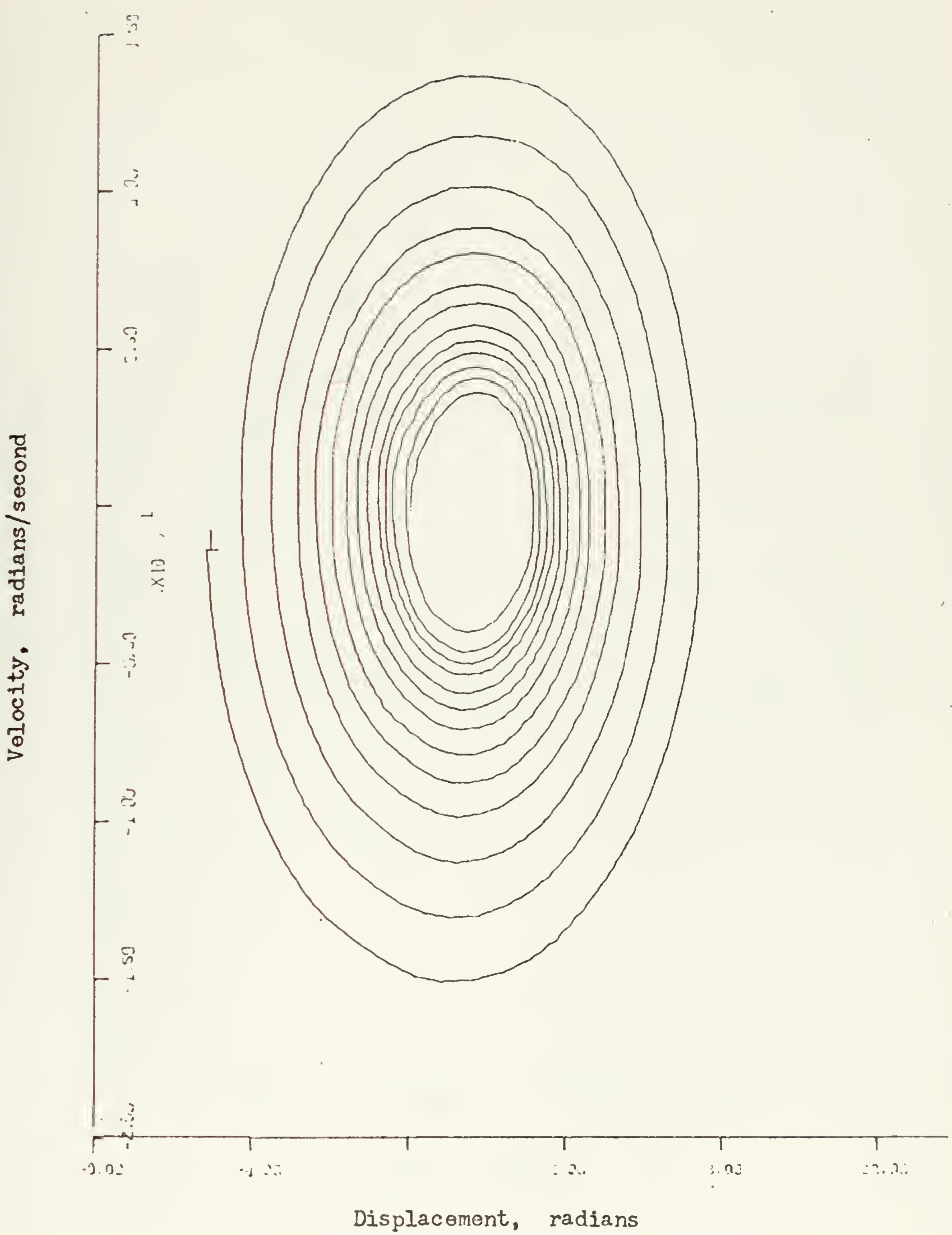



Figure 37. Third order digital servo, $K_v$=7.5, $R_n$=2, OP=10 phase plane plot

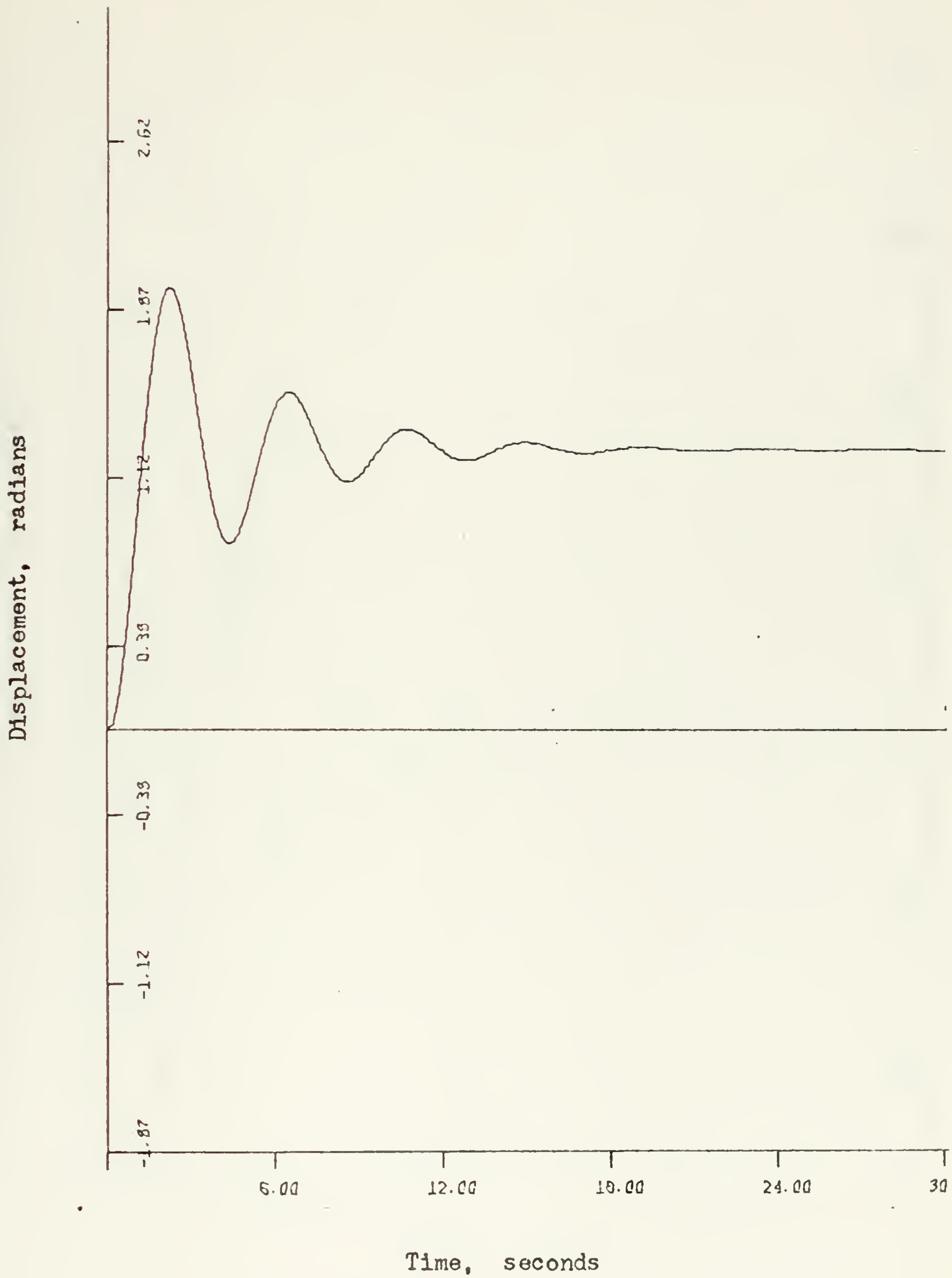



Figure 38. Third order analog servo, $K_v$=2.5, R=1.25664 displacement versus time

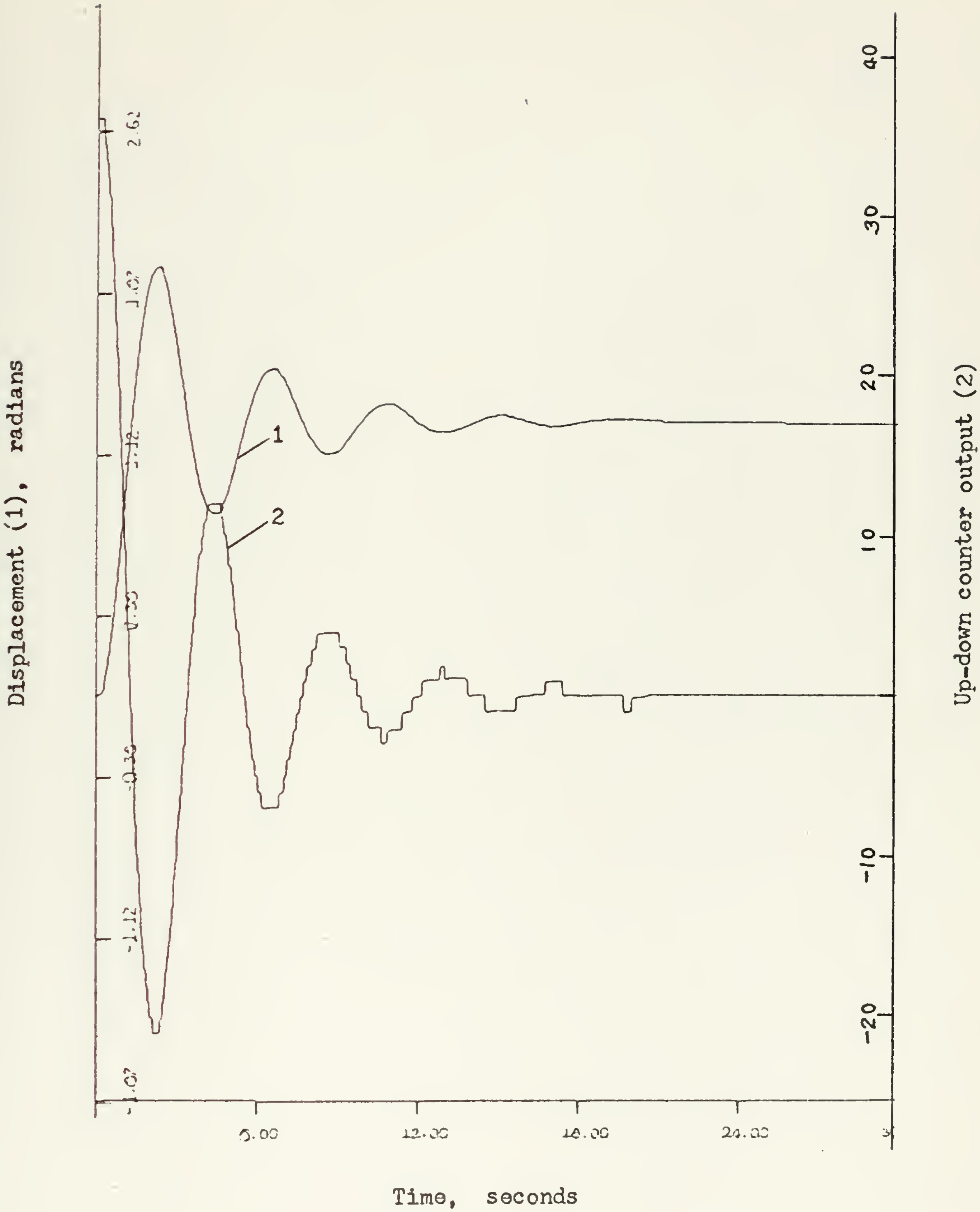



Figure 39. Third order digital servo, $K_v$=2.5, $R_n$=36, OP=180
displacement and up-down counter output vs time

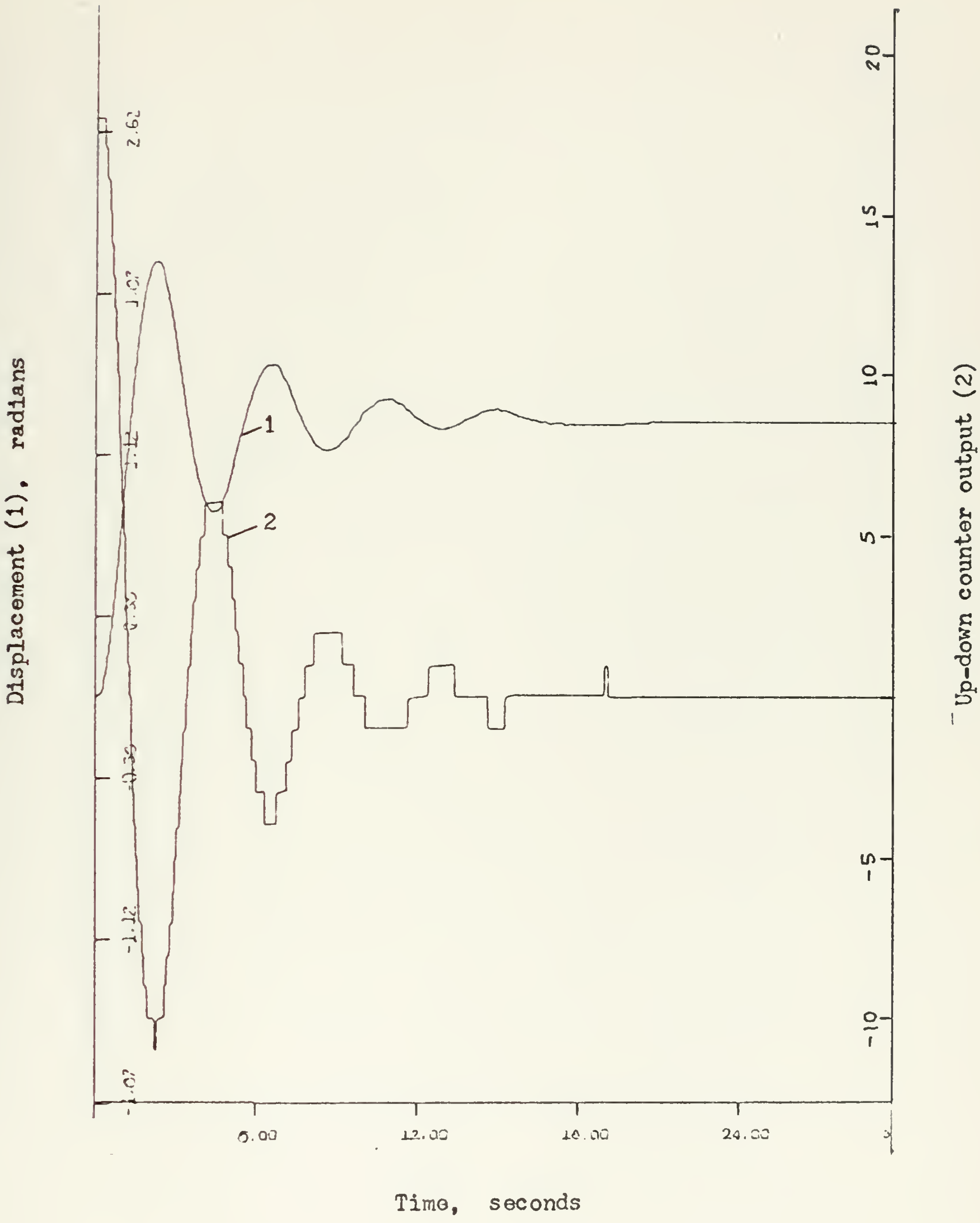



Figure 40. Third order digital servo, $K_v$=2.5, $R_n$=18, OP=90 displacement and up-down counter output vs time

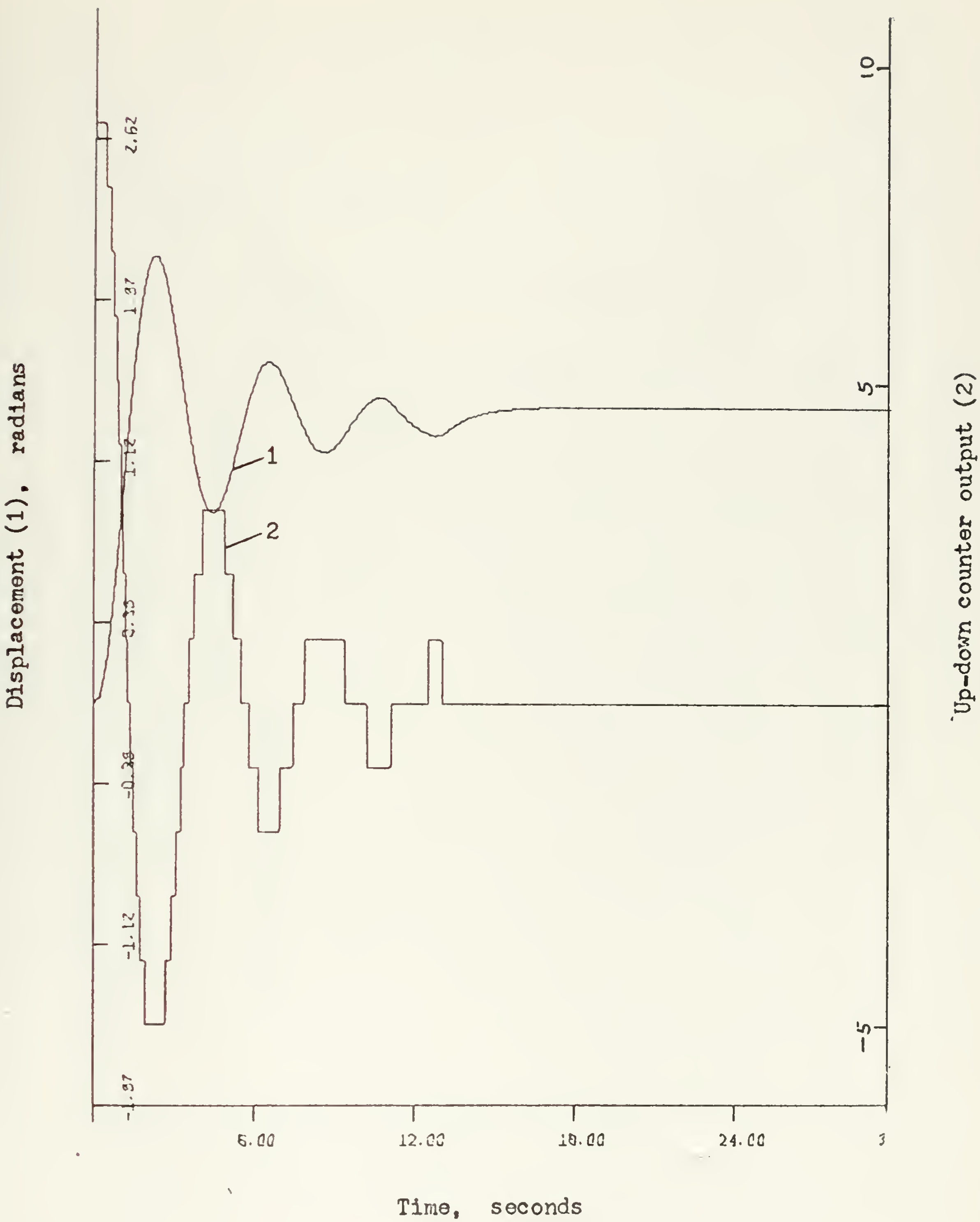



Figure 41. Third order digital servo, $K_v$=2.5, $R_n$=9, OP=45 displacement and up-down counter output vs time

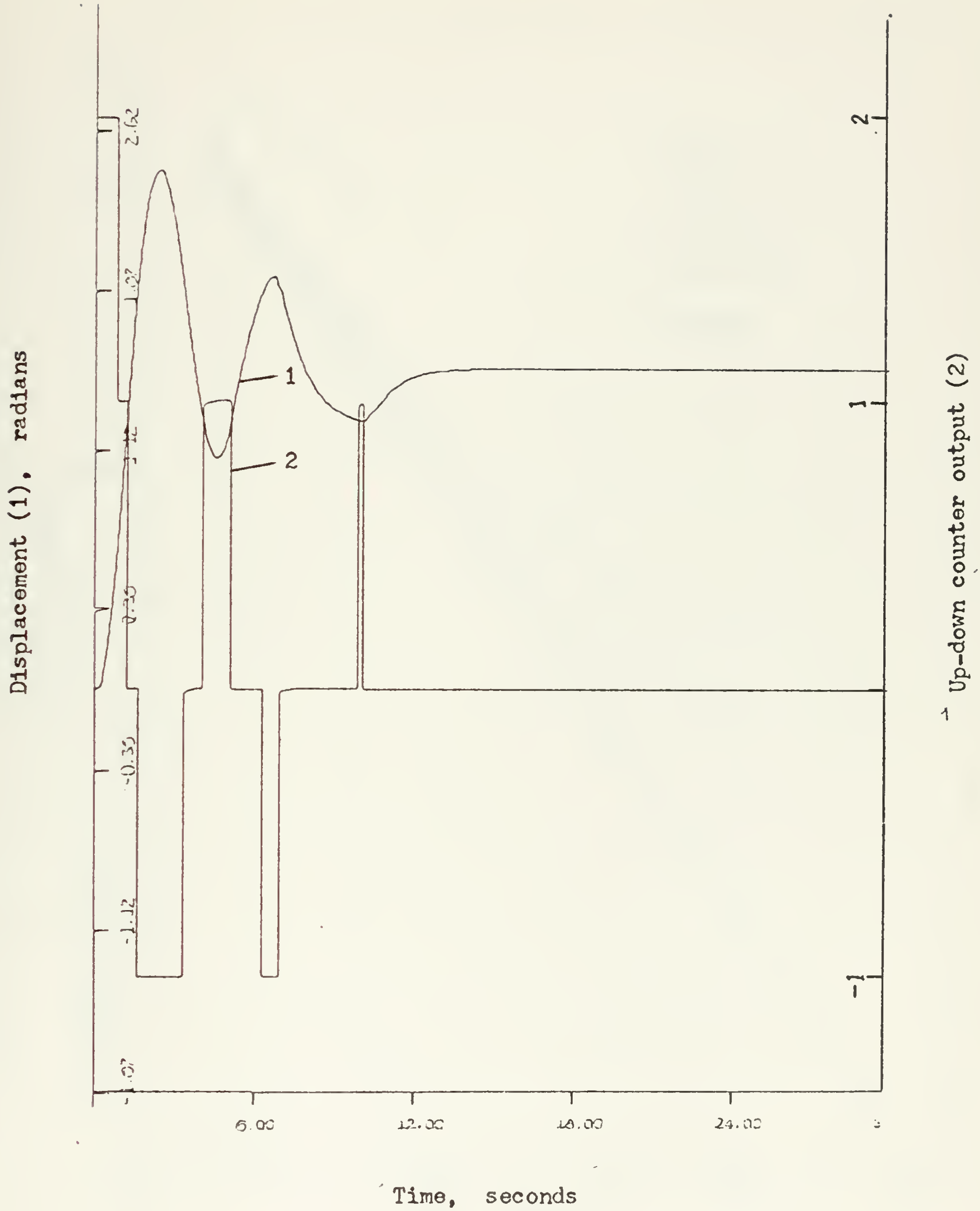



Figure 42. Third order digital servo, $K_v=2.5$, $R_n=2$, OP=10 displacement and up-down counter output vs time

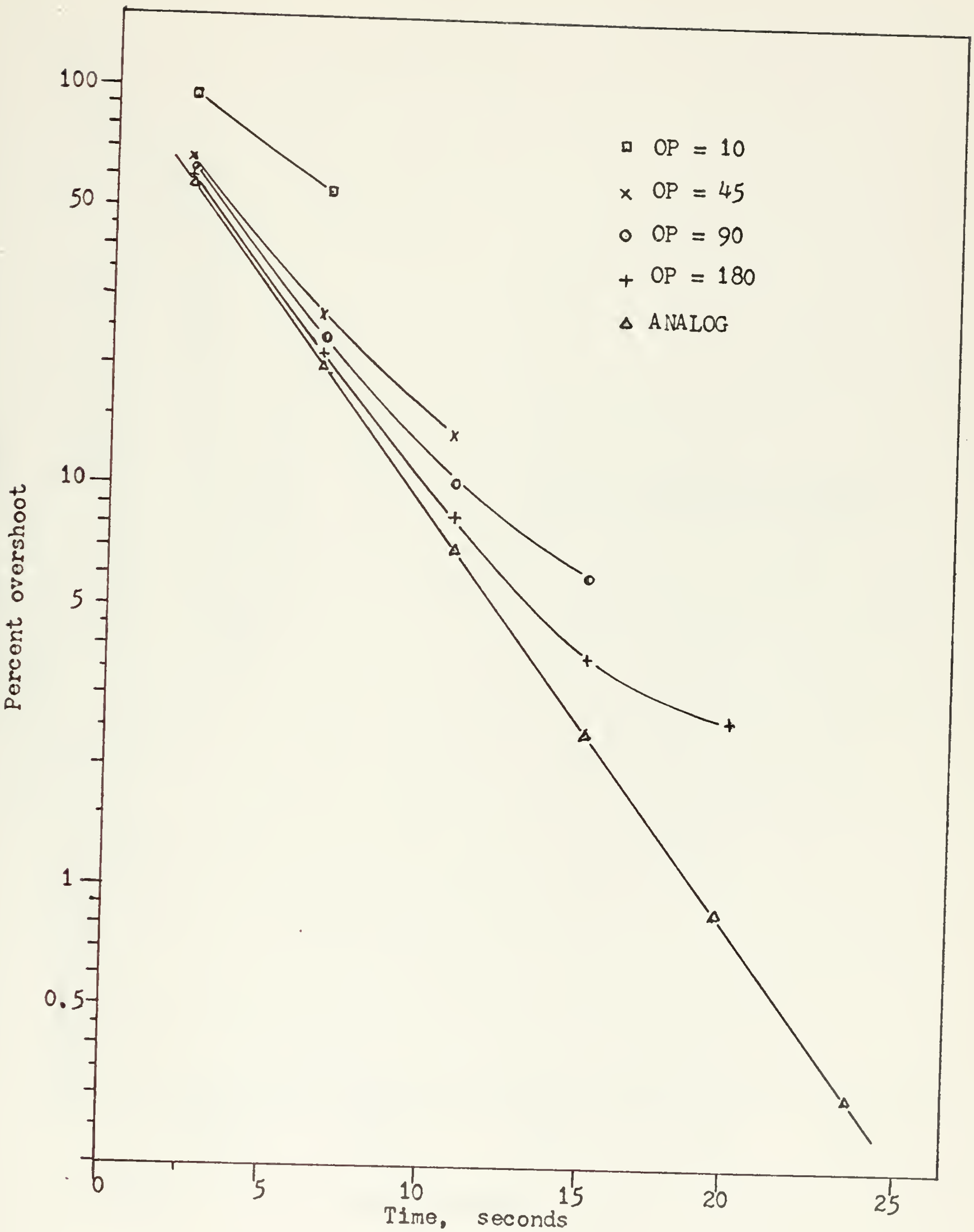



Figure 43. Third order servo system, $K_v$=2.5
OP=10, 45, 90, 180, and analog
percent overshoot versus time

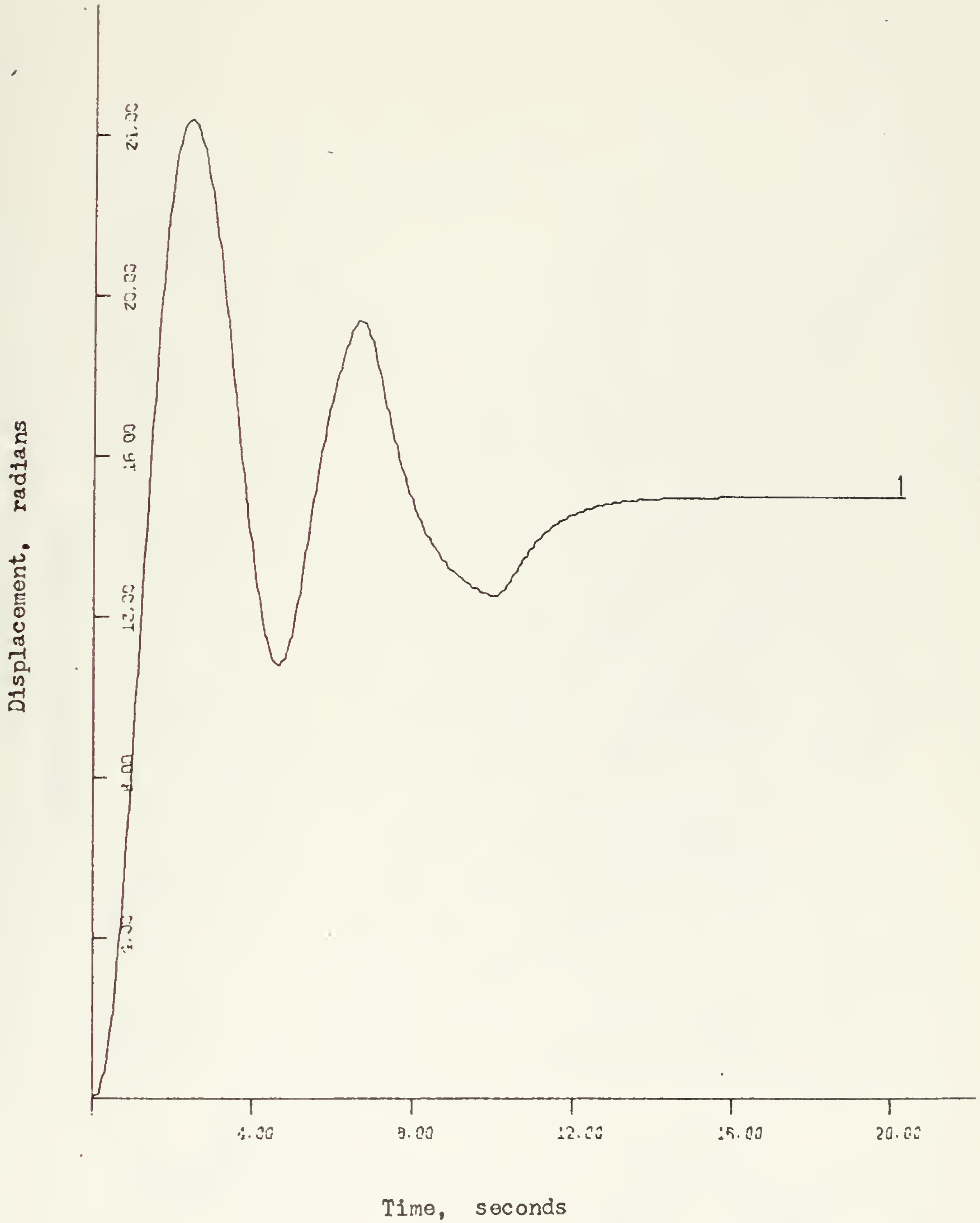



Figure 44. Third order digital servo, $K_v$=2.5, $R_n$=2, OP=1
displacement versus time


Figure 45. Third order digital servo, displacement vs time ramp input, B=0.5, 1.0, 1.5, and 2.0, OP=10, $K_v$=2.5

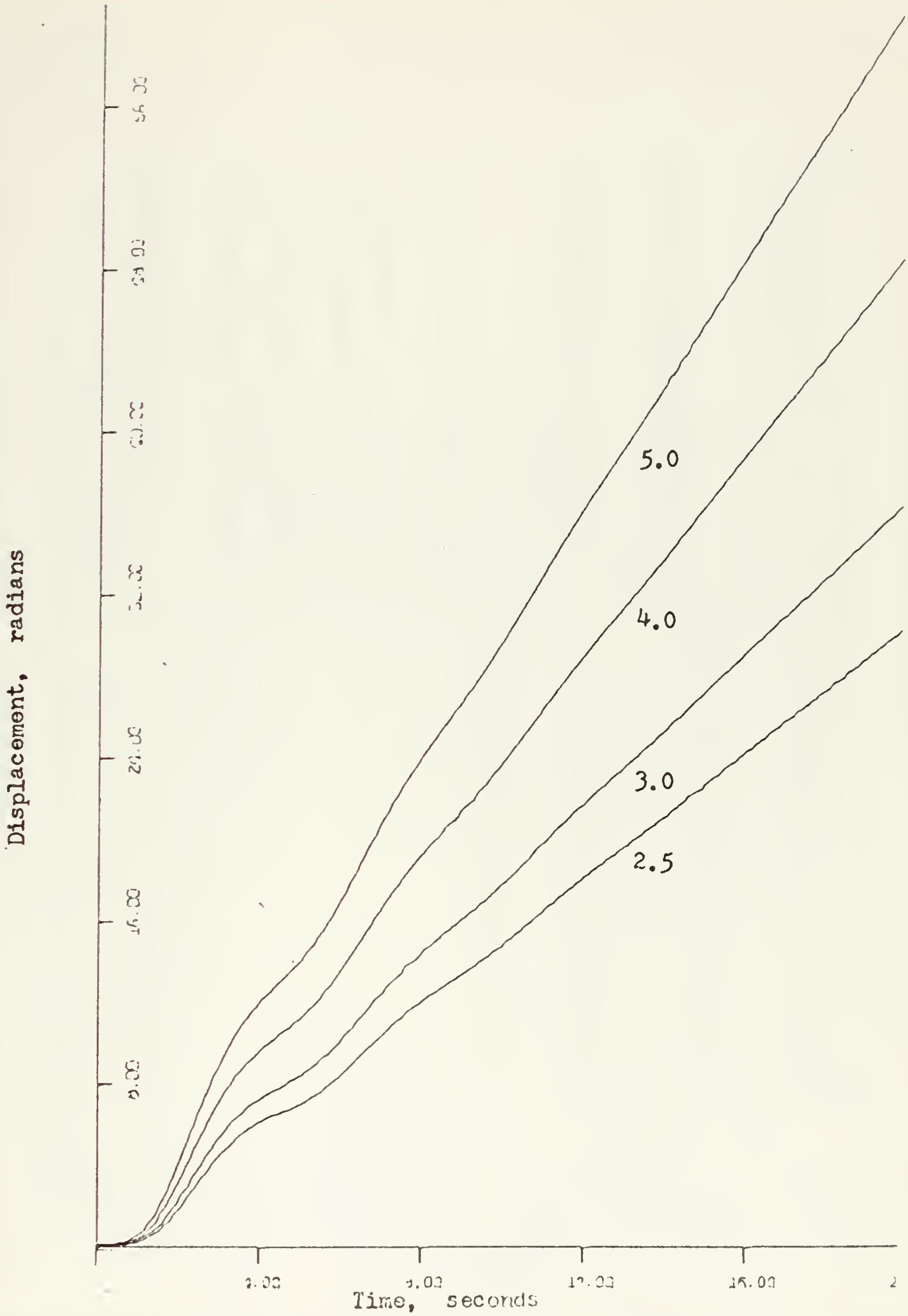



Figure 46. Third order digital servo, displacement vs time ramp input, B =2.5, 3.0, 4.0, and 5.0, OP=10, $K_v$=2.5

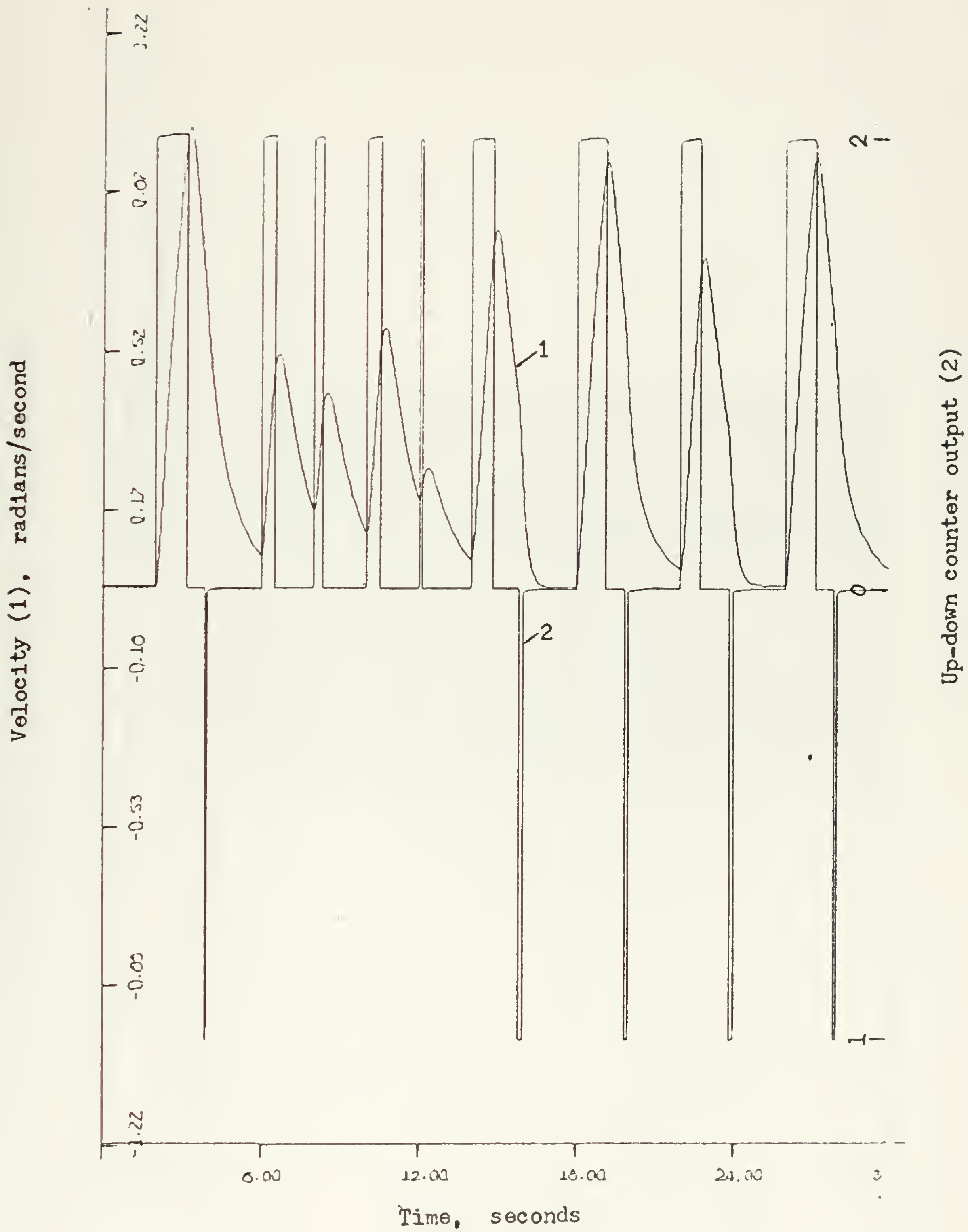



Figure 47. Third order digital servo, ramp input, B=0.5, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

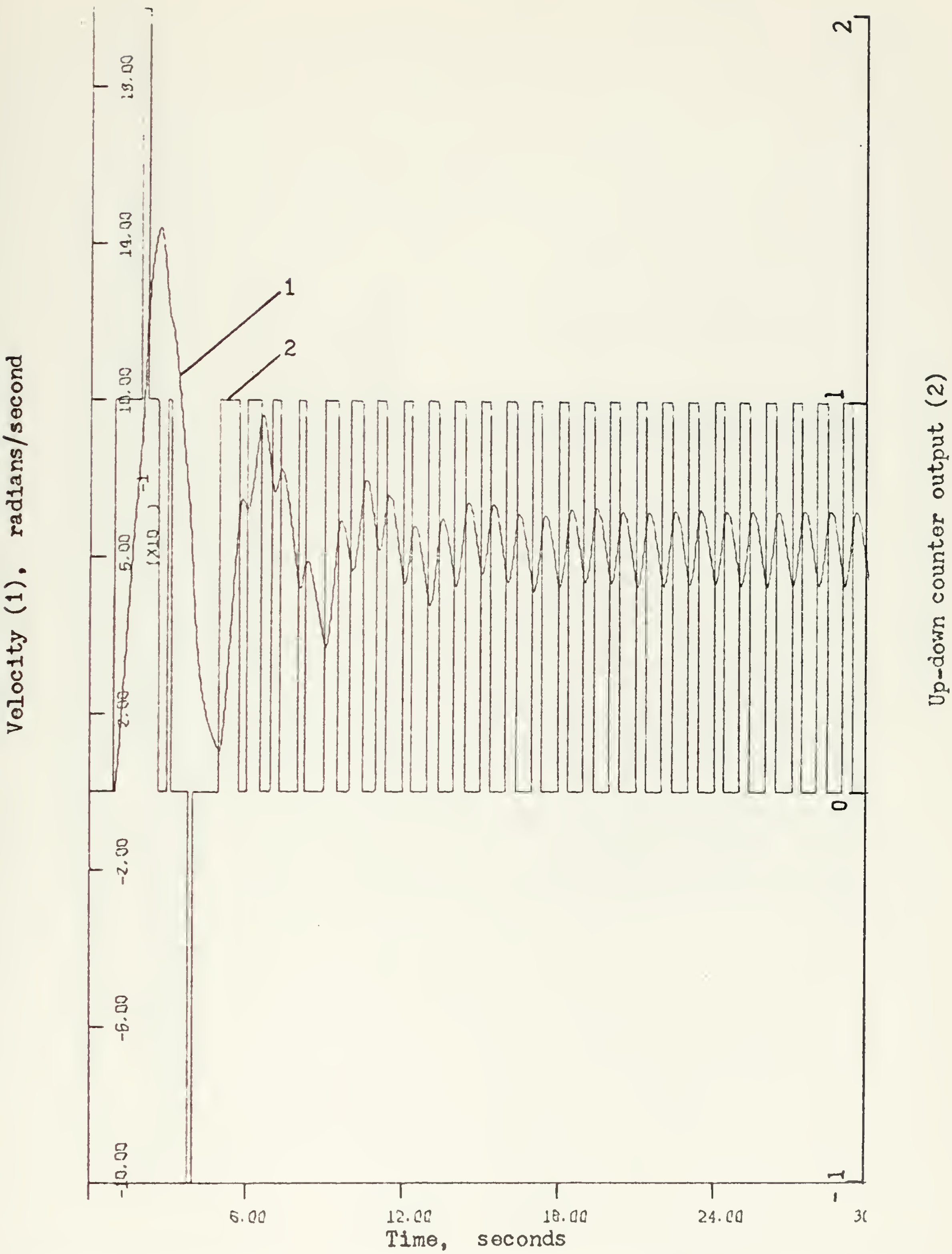



Figure 48. Third order digital servo, ramp input, B=1.0, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

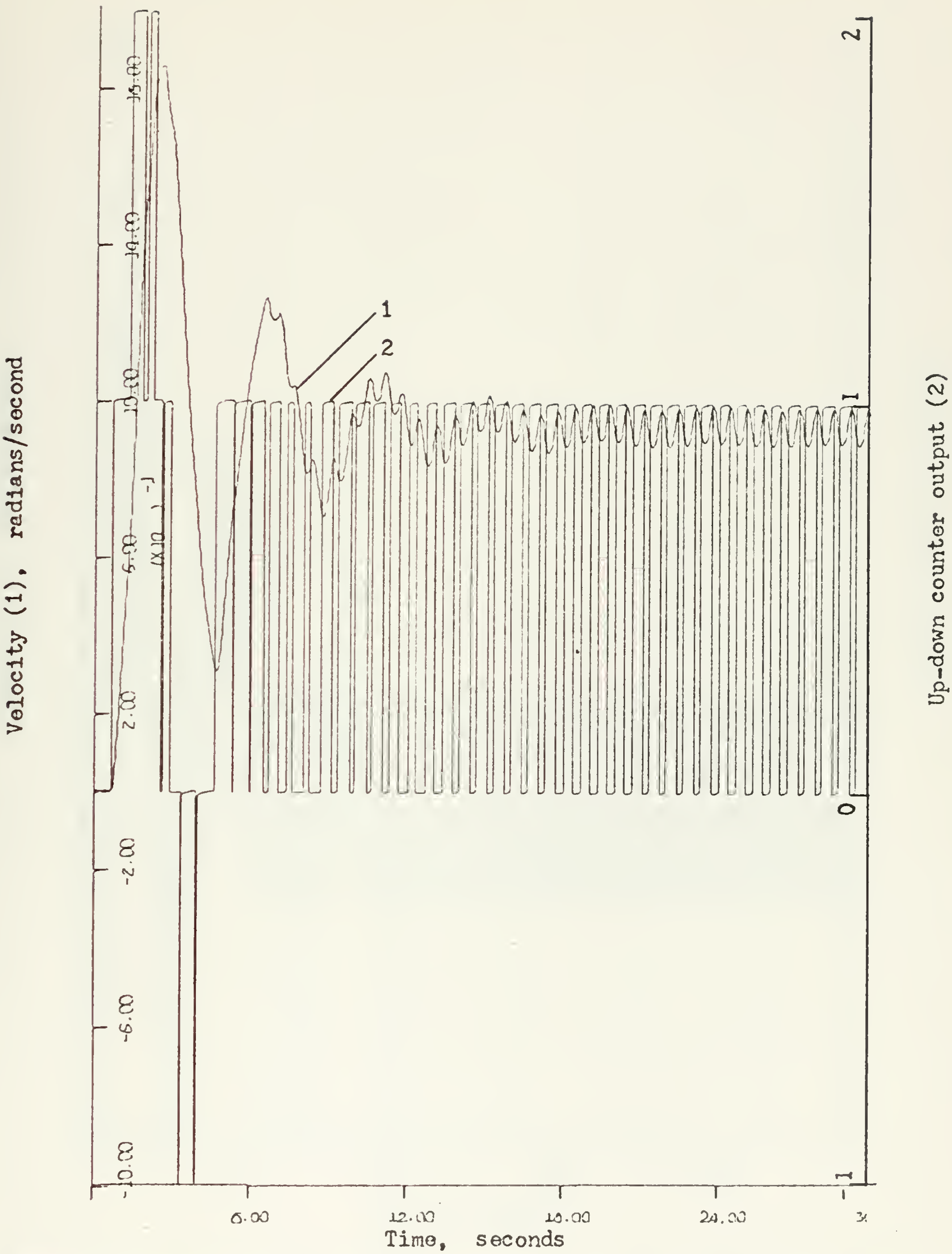



Figure 49. Third order digital servo, ramp input, B =1.5, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

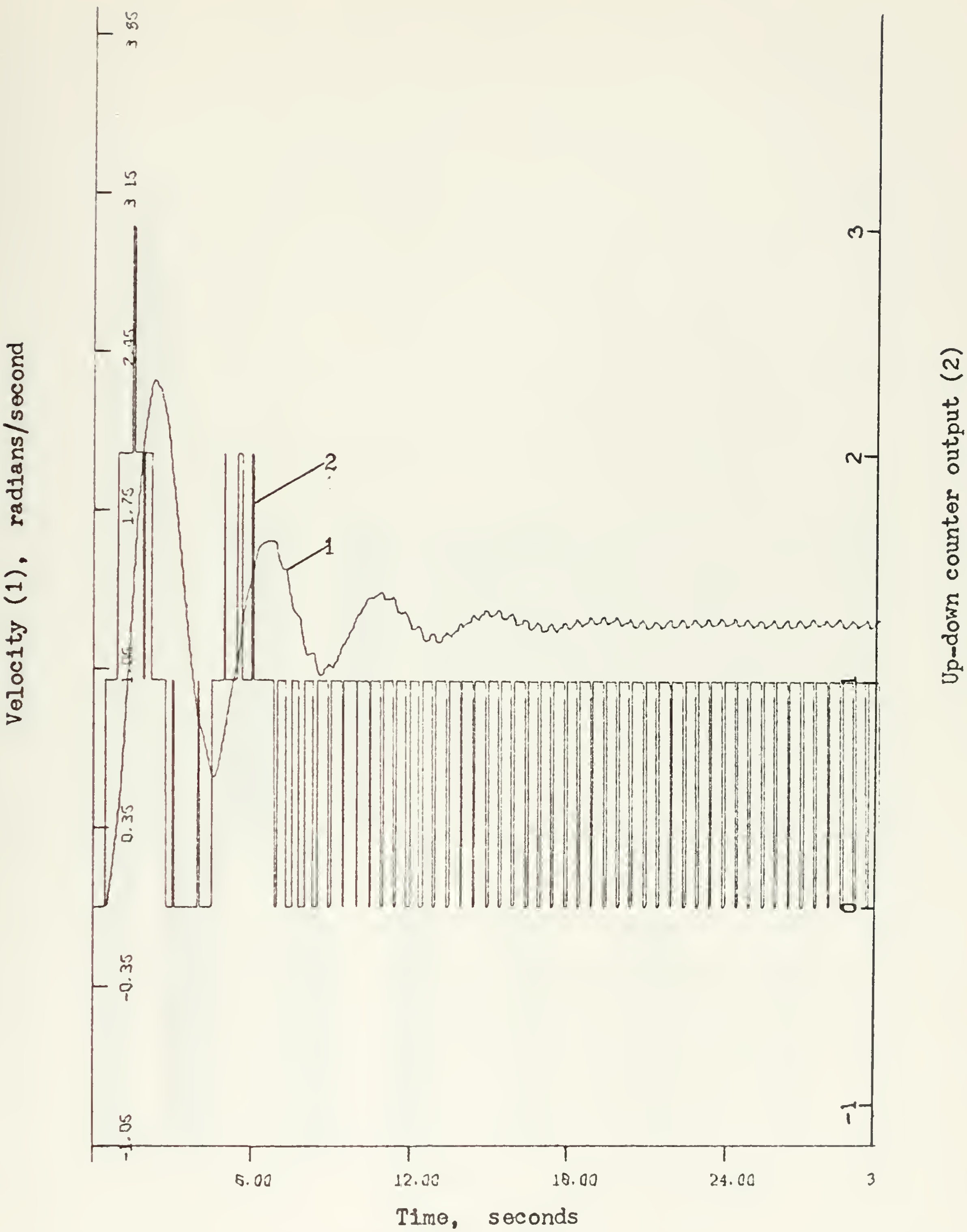



Figure 50. Third order digital servo, ramp input, B=2.0, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

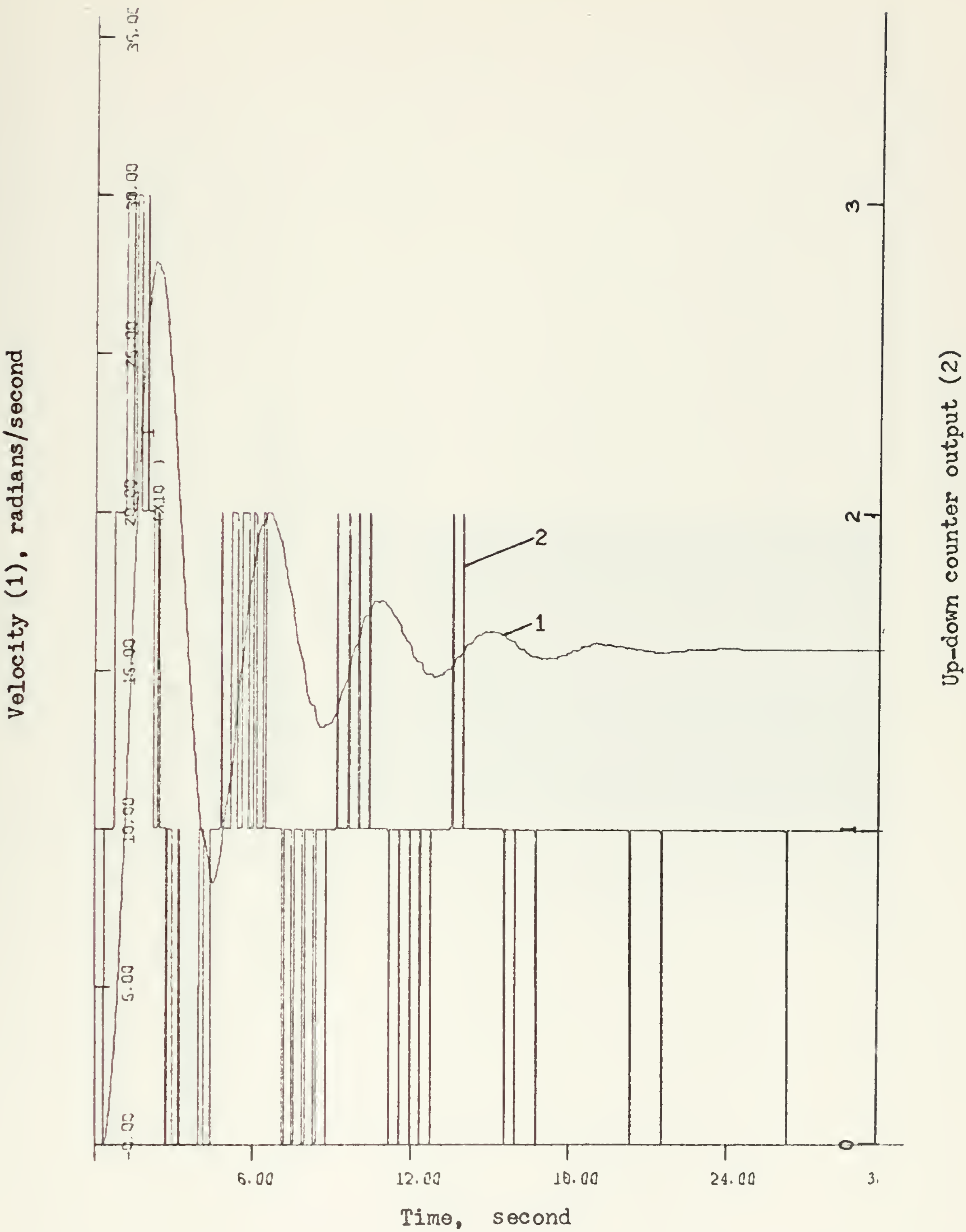



Figure 51. Third order digital servo, ramp input, B=2.5, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

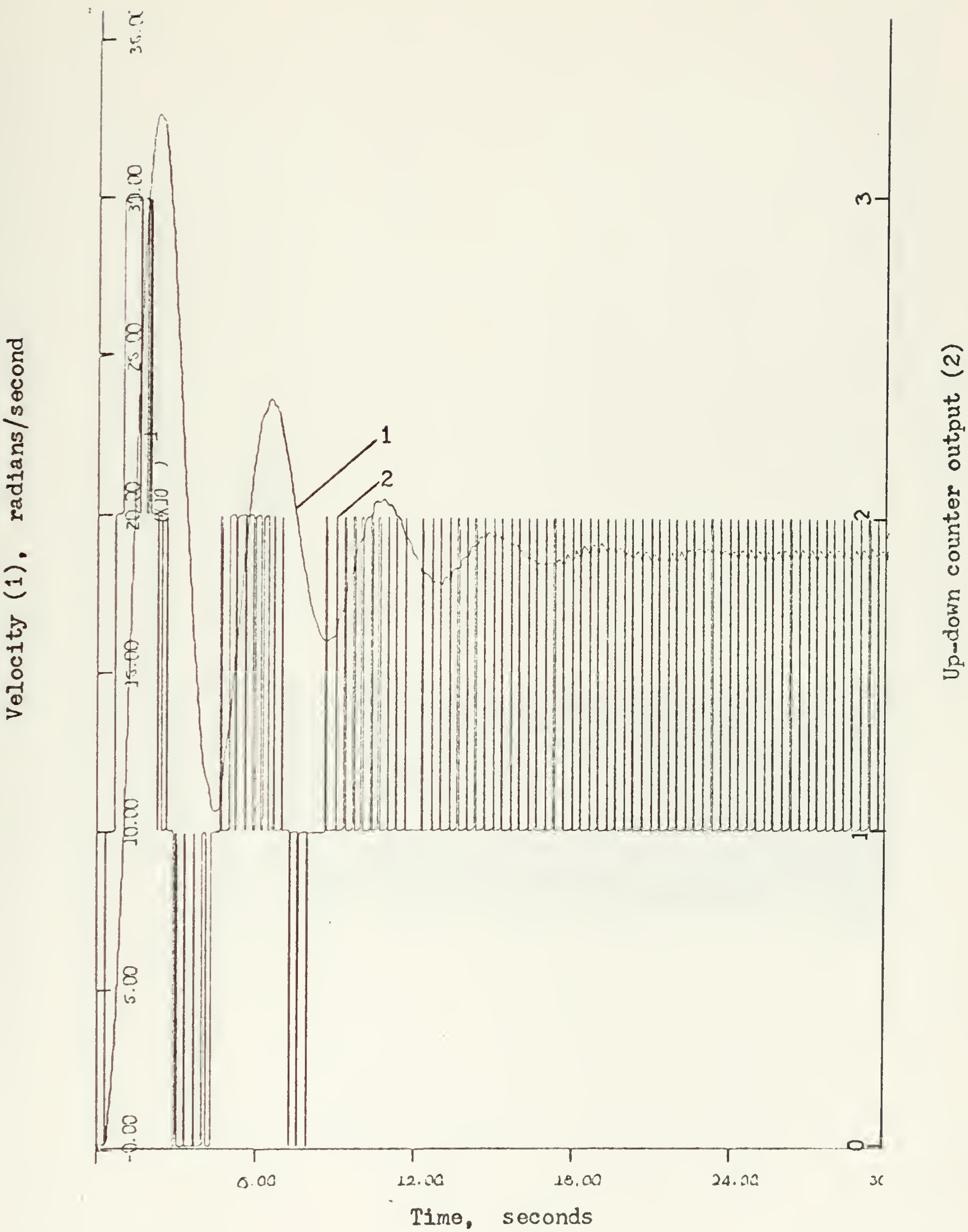



Figure 52. Third order digital servo, ramp input, B =3.0, $K_v$=2.5, OP=10
velocity and up-down counter output versus time


Figure 53. Third order digitalservo, ramp input, B=4.0, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

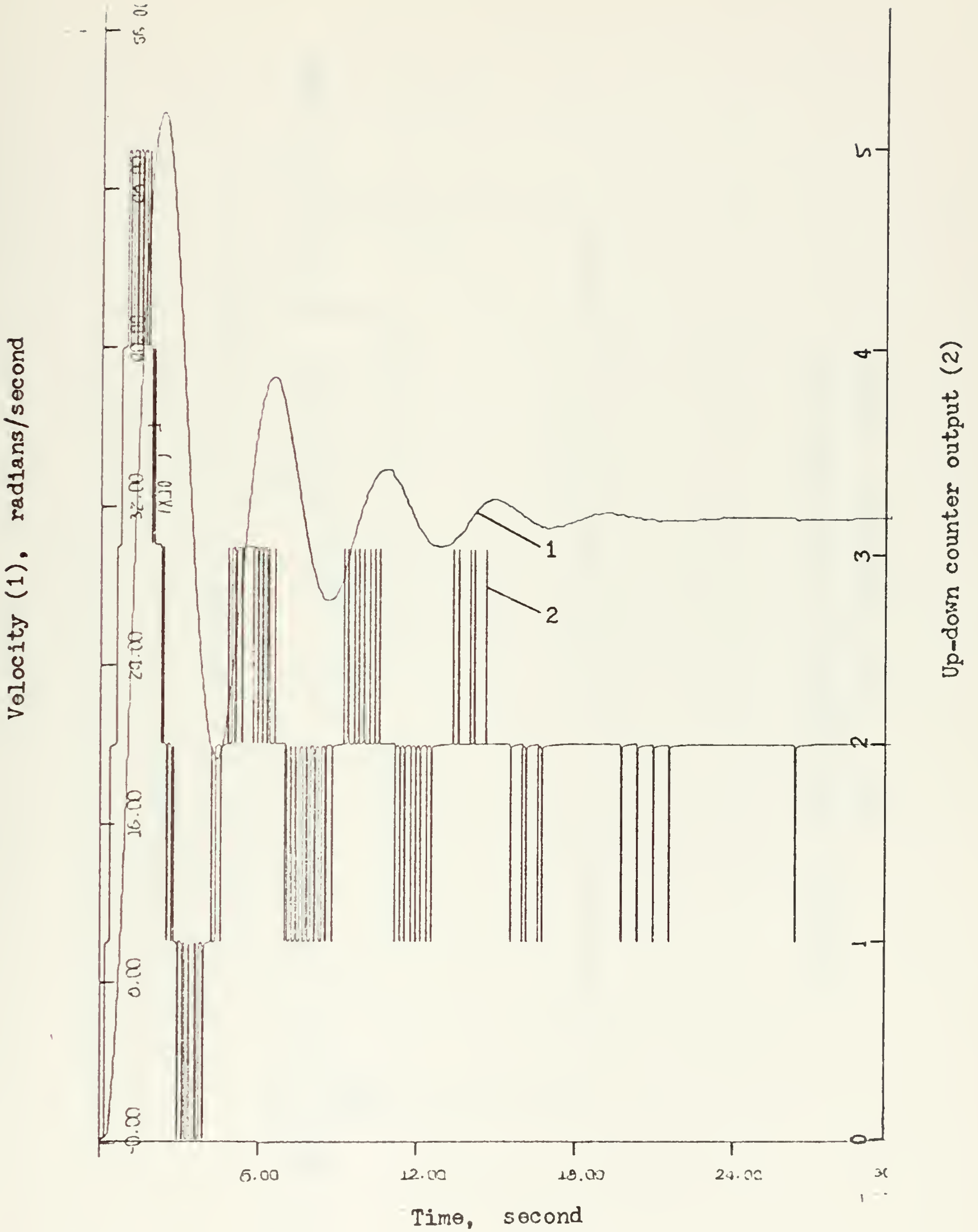



Figure 54. Third order digital servo, ramp input, B=5.0, $K_v$=2.5, OP=10 velocity and up-down counter output versus time

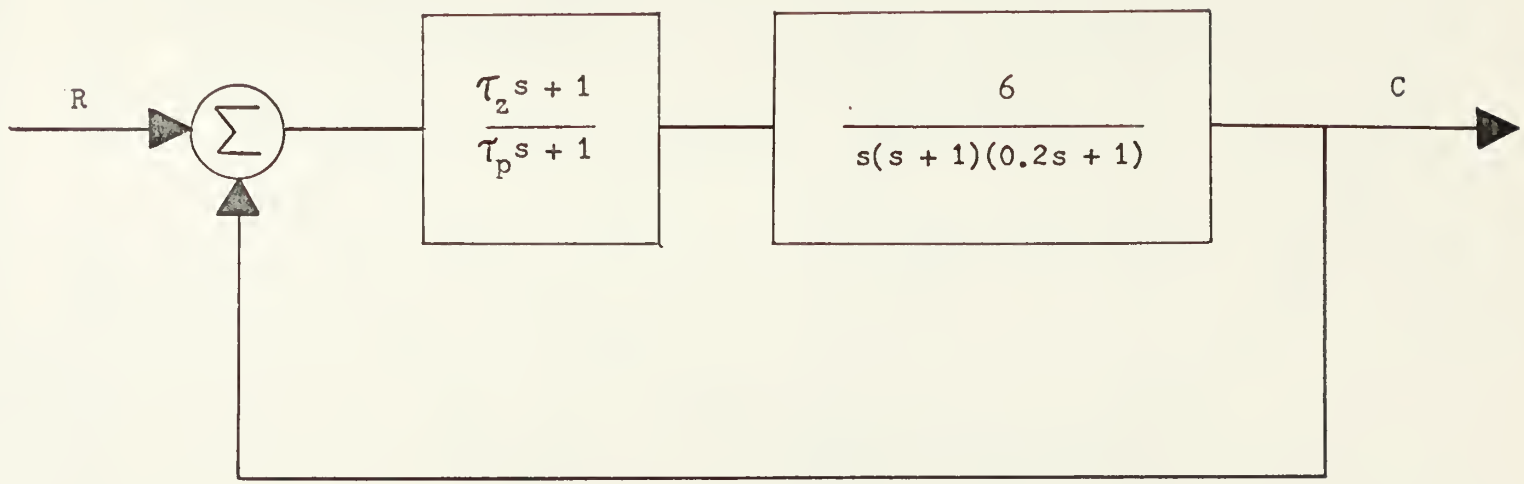



Figure 55. Block diagram of a third order servo with a lag compensation section

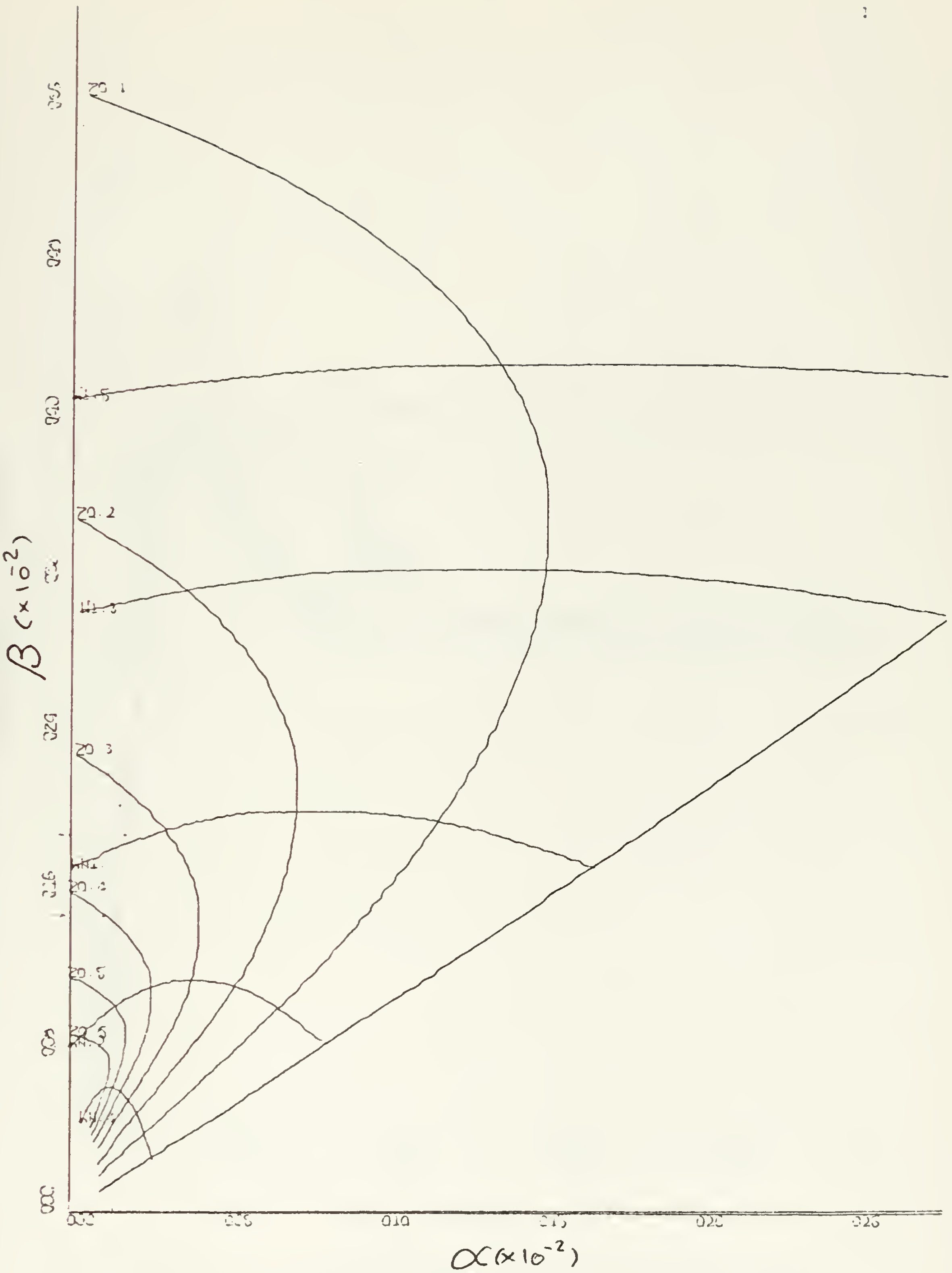



Figure 56. Parameter plane plot for lag compensation of the third order positioning servo

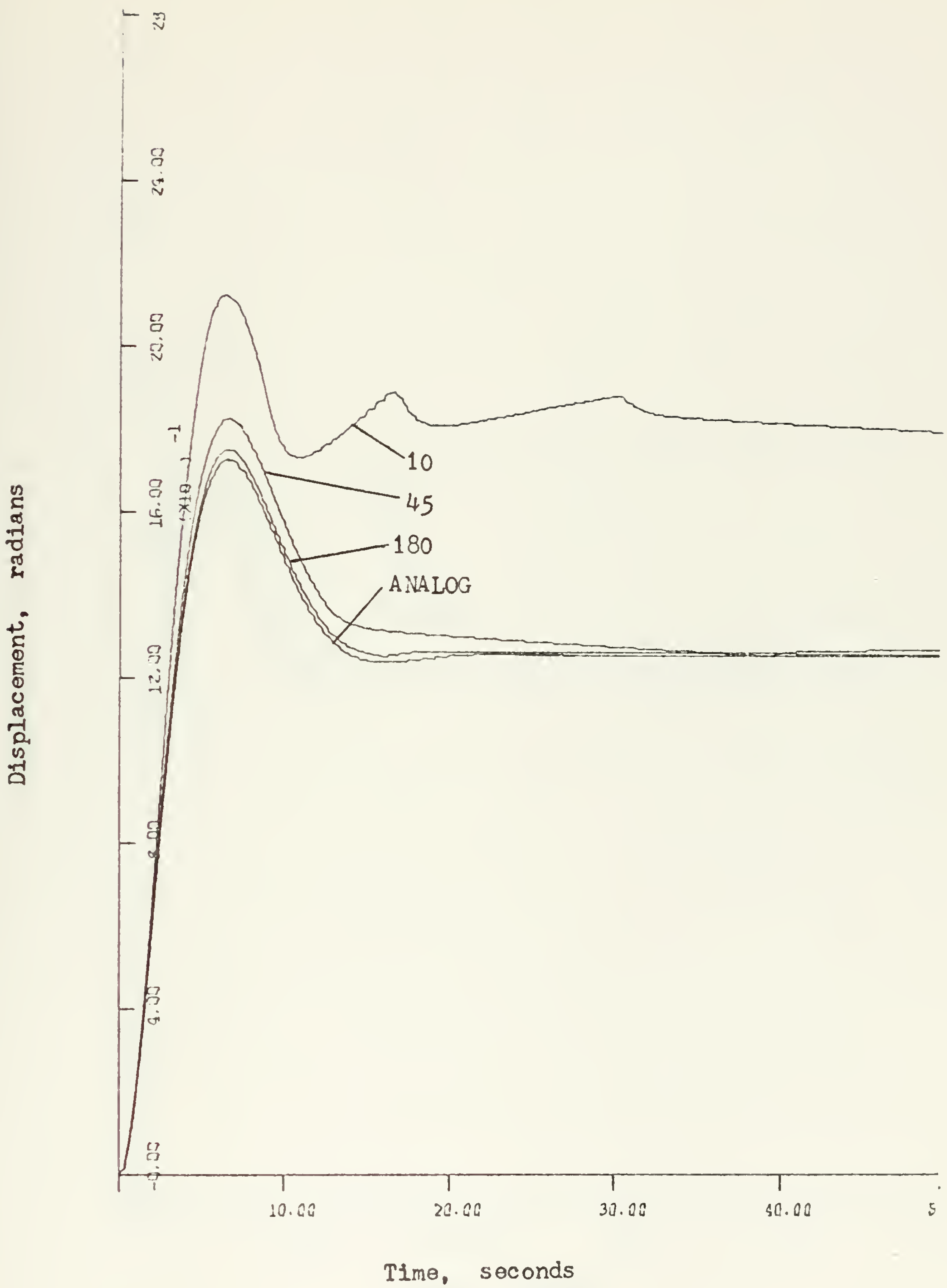



Figure 57. Lag compensated third order positioning servo
p=0.013, z=0.18, OP=10, 45, 180, and analog

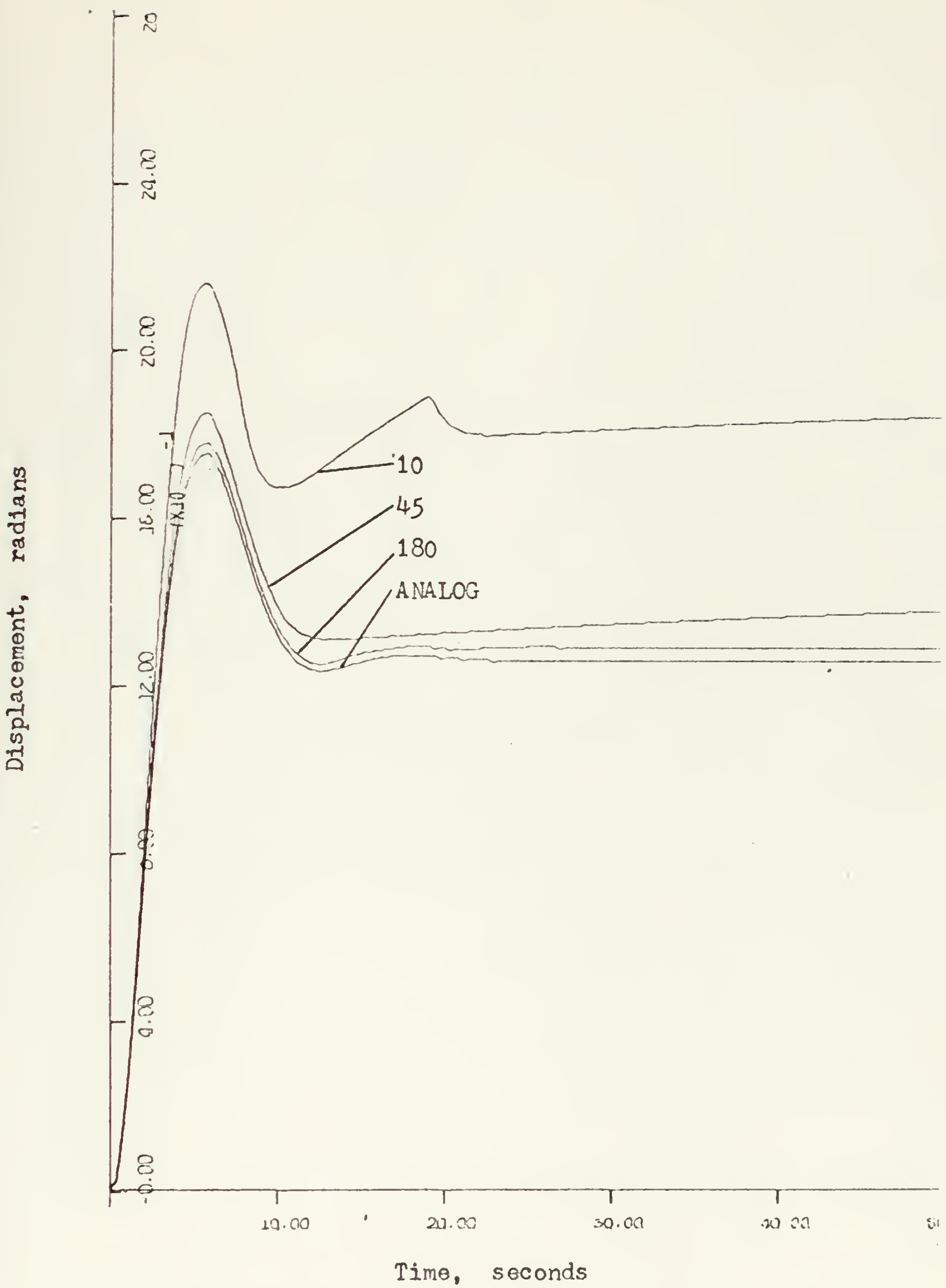



Figure 58. Lag compensated third order positioning servo
p=0.017, z=0.19, OP=10, 45, 180, and analog

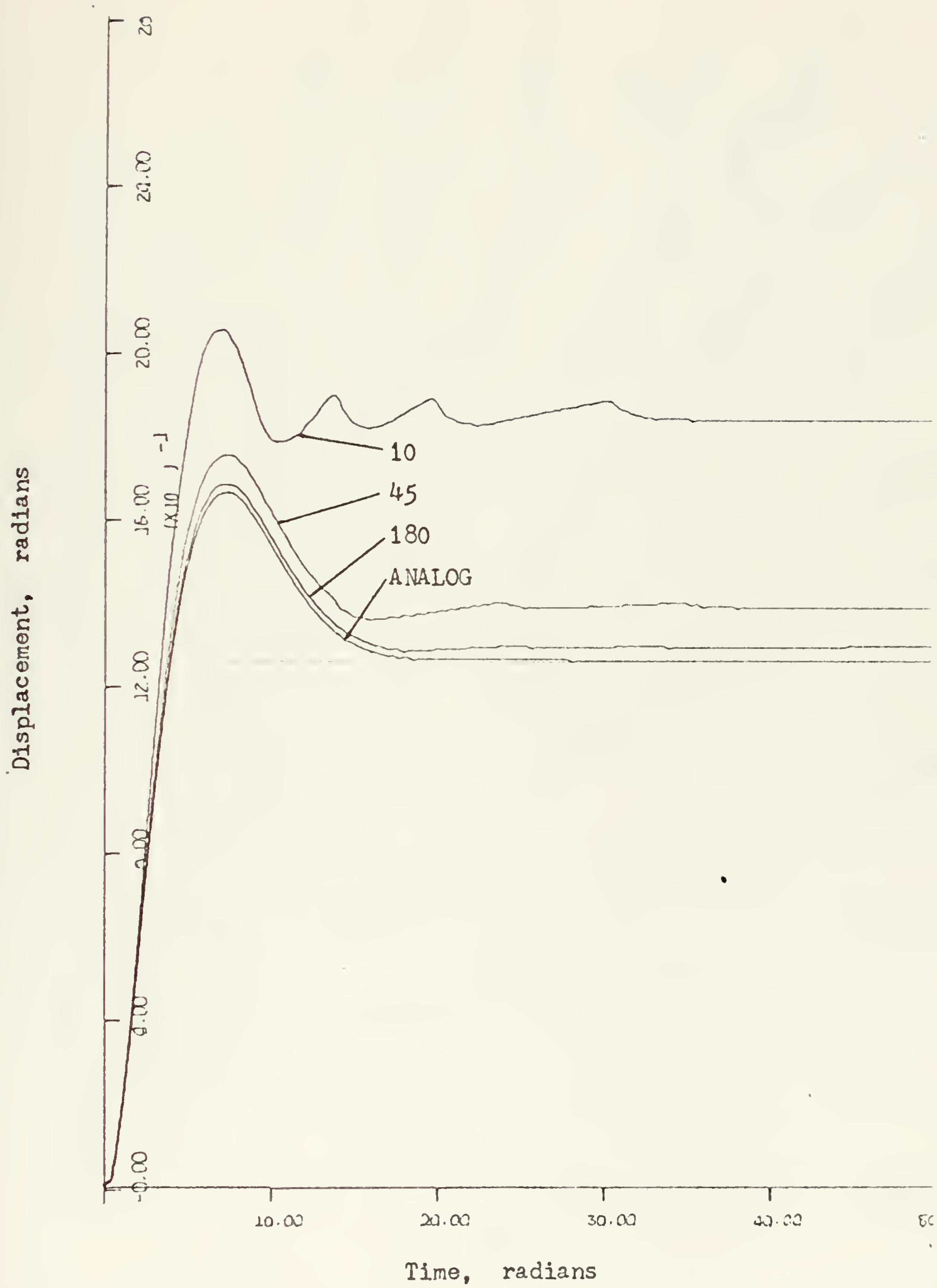



Figure 59. Lag compensated third order positioning servo
p=0.01, z=0.15, OP=10, 45, 180, and analog

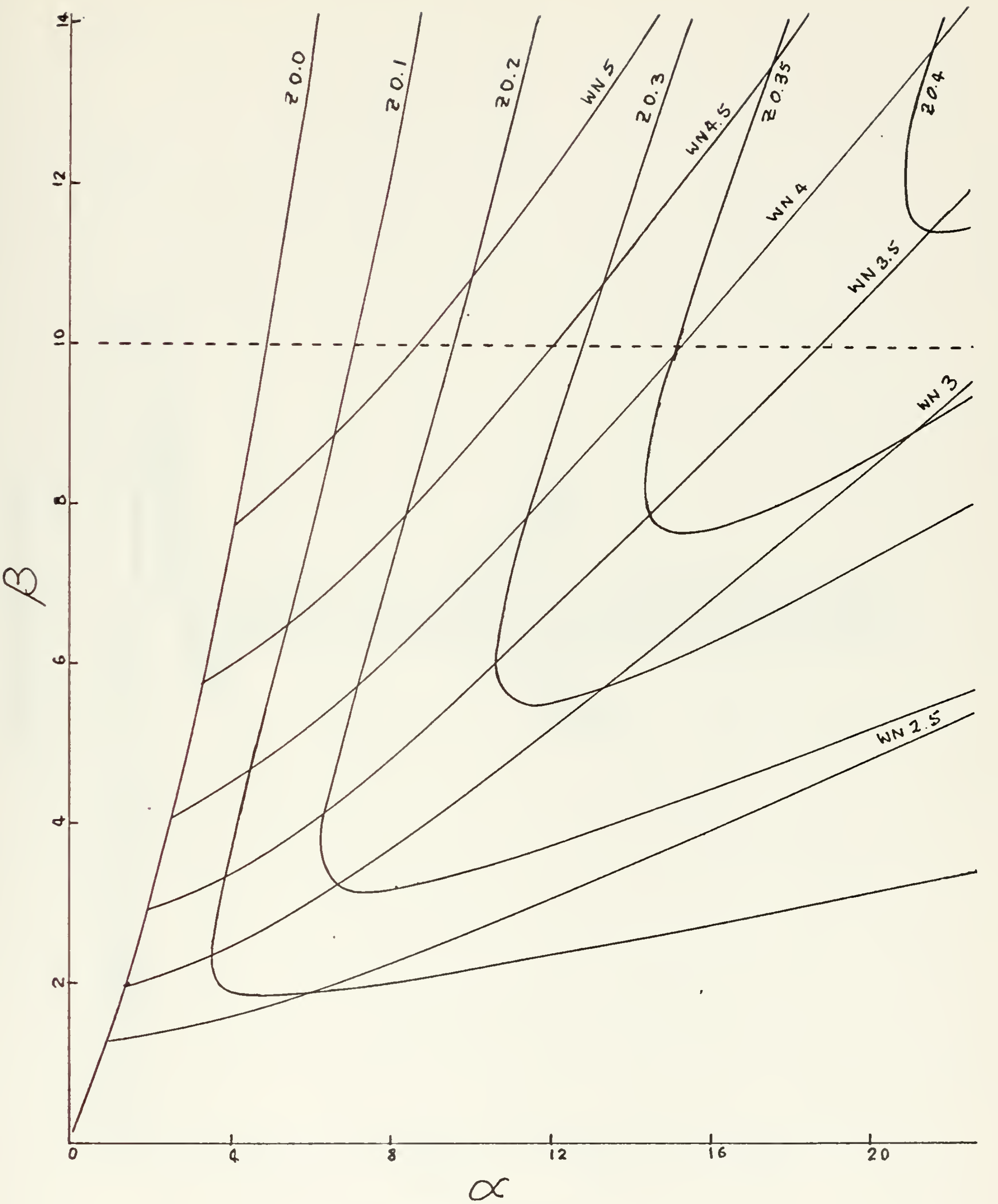



Figure 60. Parameter plane plot for lead compensation of the third order positioning servo

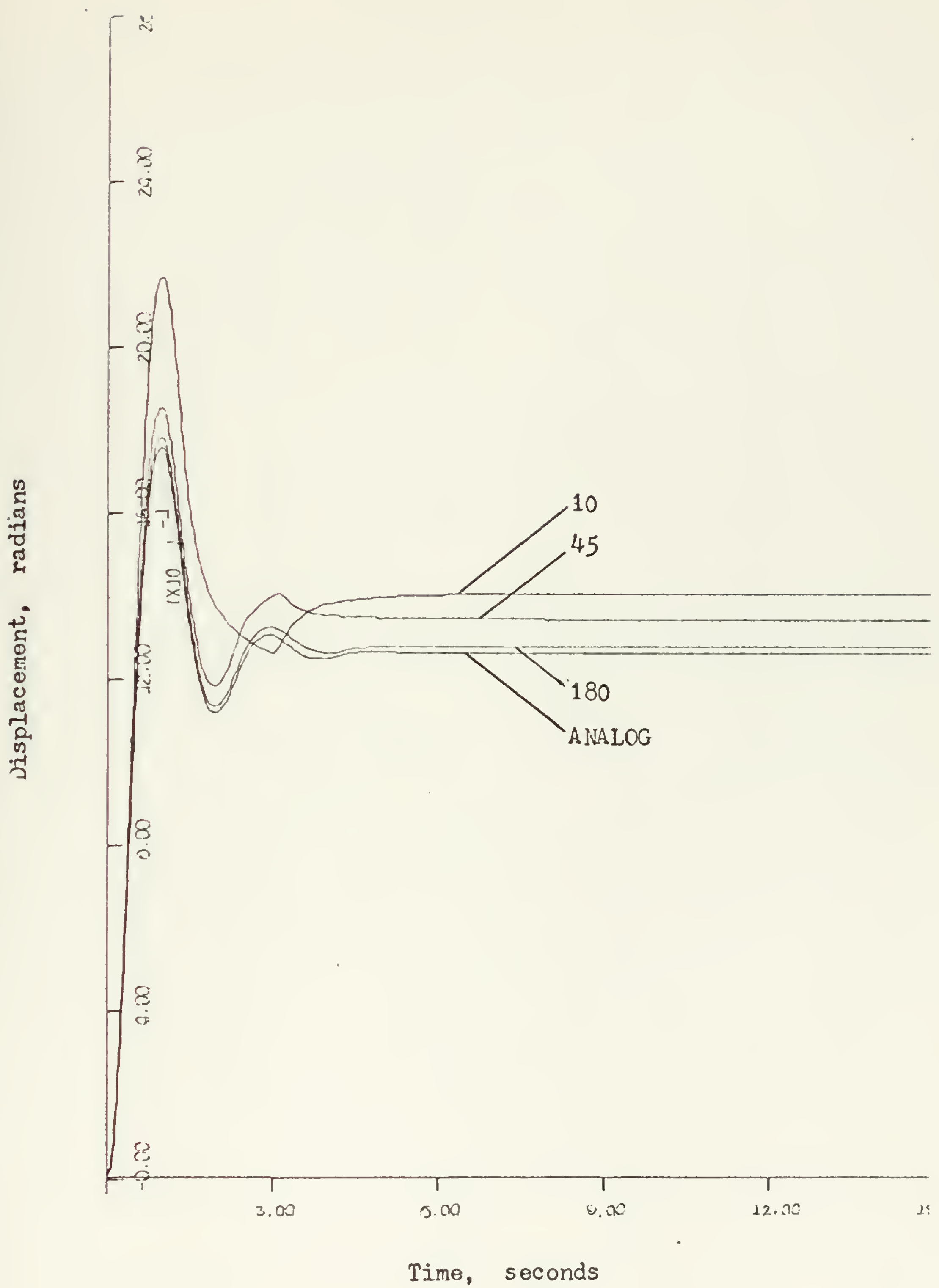



Figure 61. Lead compensated third order positioning servo
p=14.5, z=1.85, OP=10, 45, 180, and analog

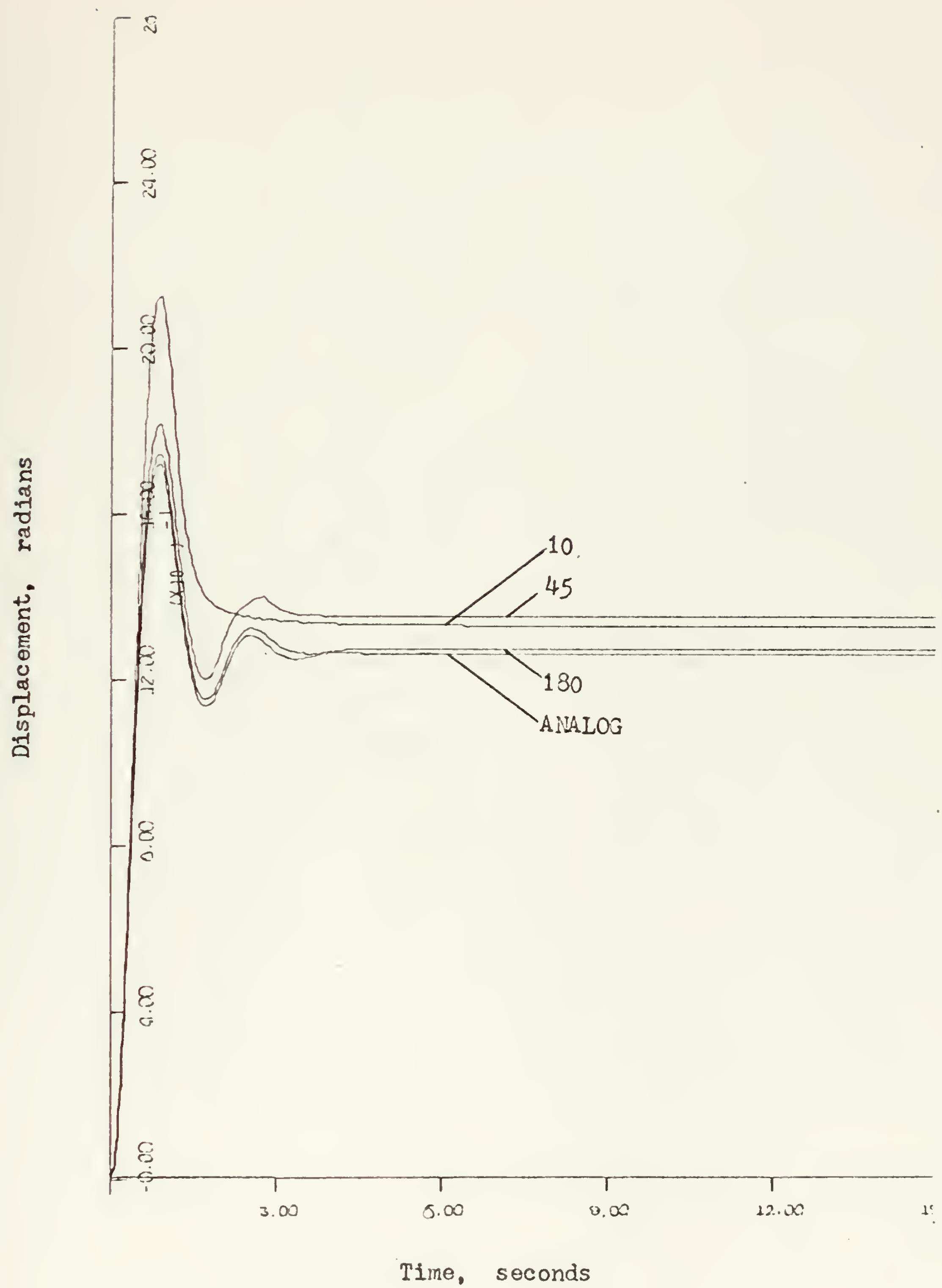



Figure 62. Lead compensated third order positioning servo
p=15.0, z=1.50, OP=10, 45, 180, and analog

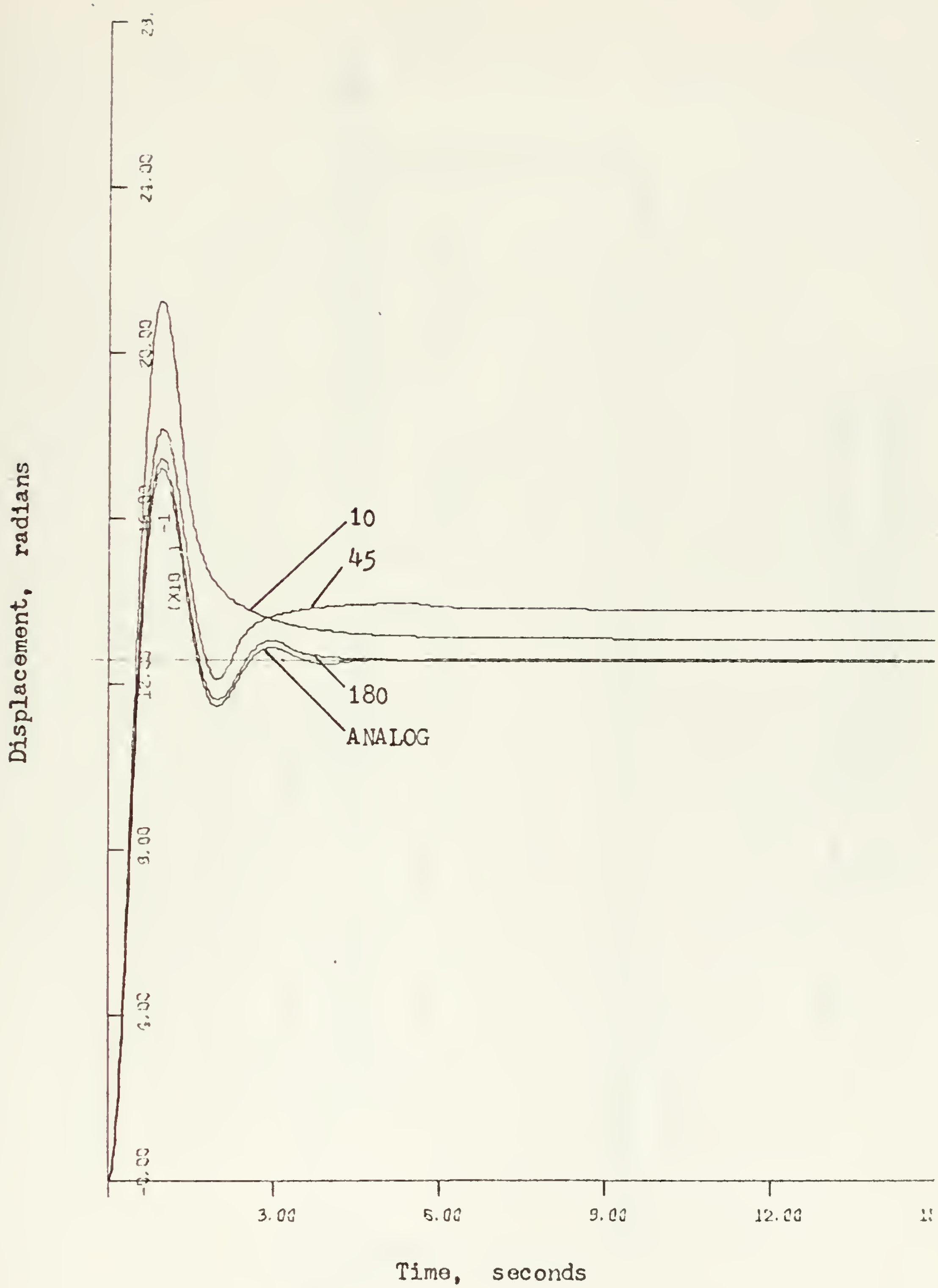



Figure 63. Lead compensated third order positioning servo
$p=19.0$, $z=1.90$, OF=10, 45, 180, and analog

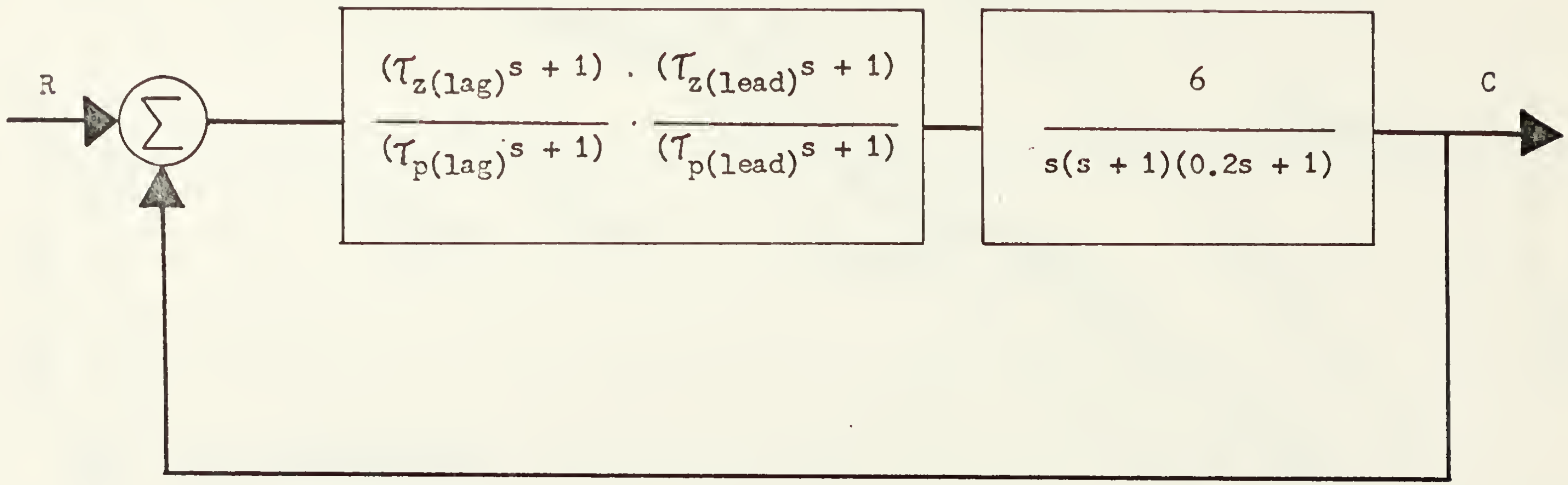



Figure 64. Block diagram of a third order servo with a lag-lead compensation section

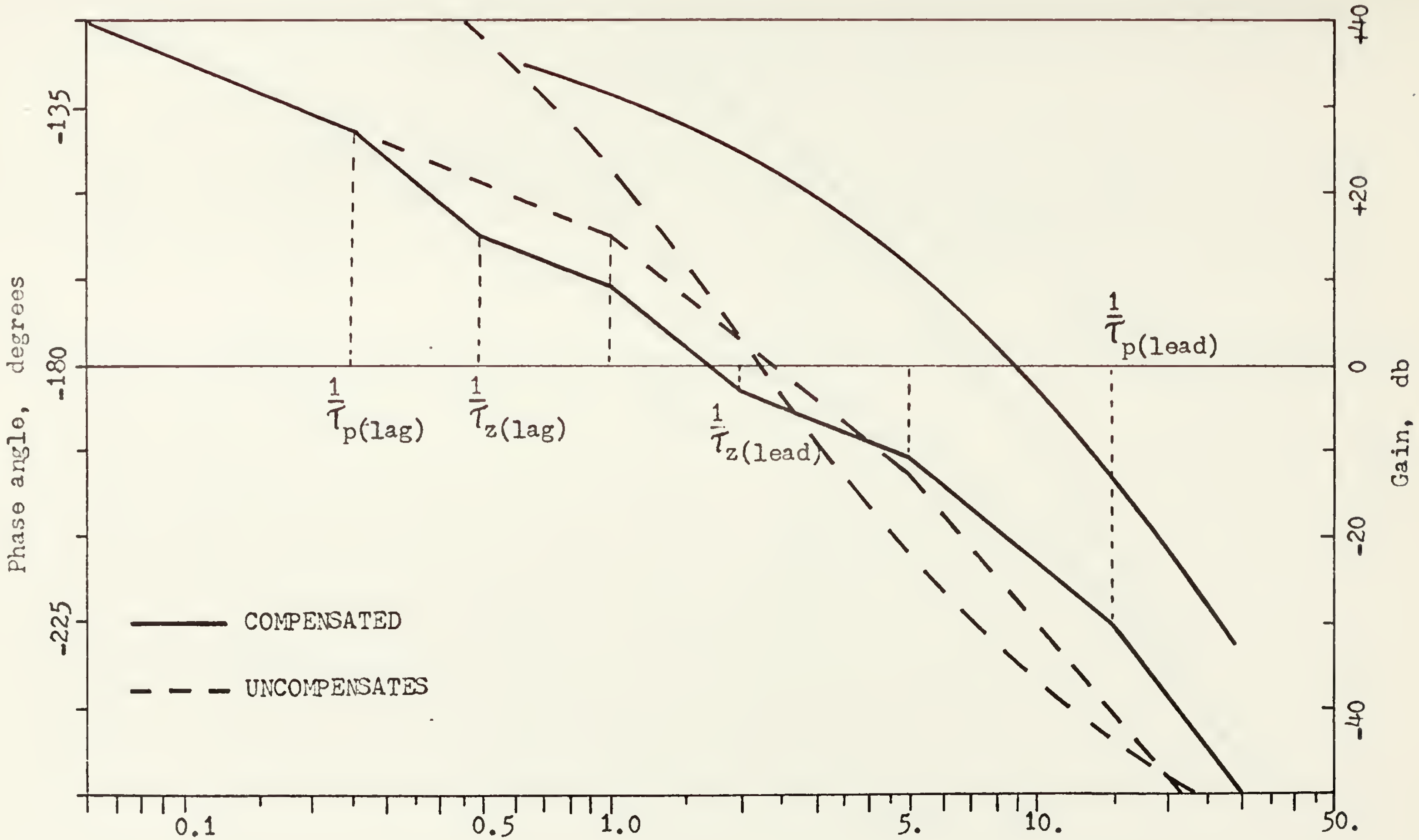



Figure 65. Lag-lead compensated third order servo

$p_{lag}=0.25$, $z_{lag}=0.5$, $p_{lead}=20$, $z_{lead}=2$

Bode plot

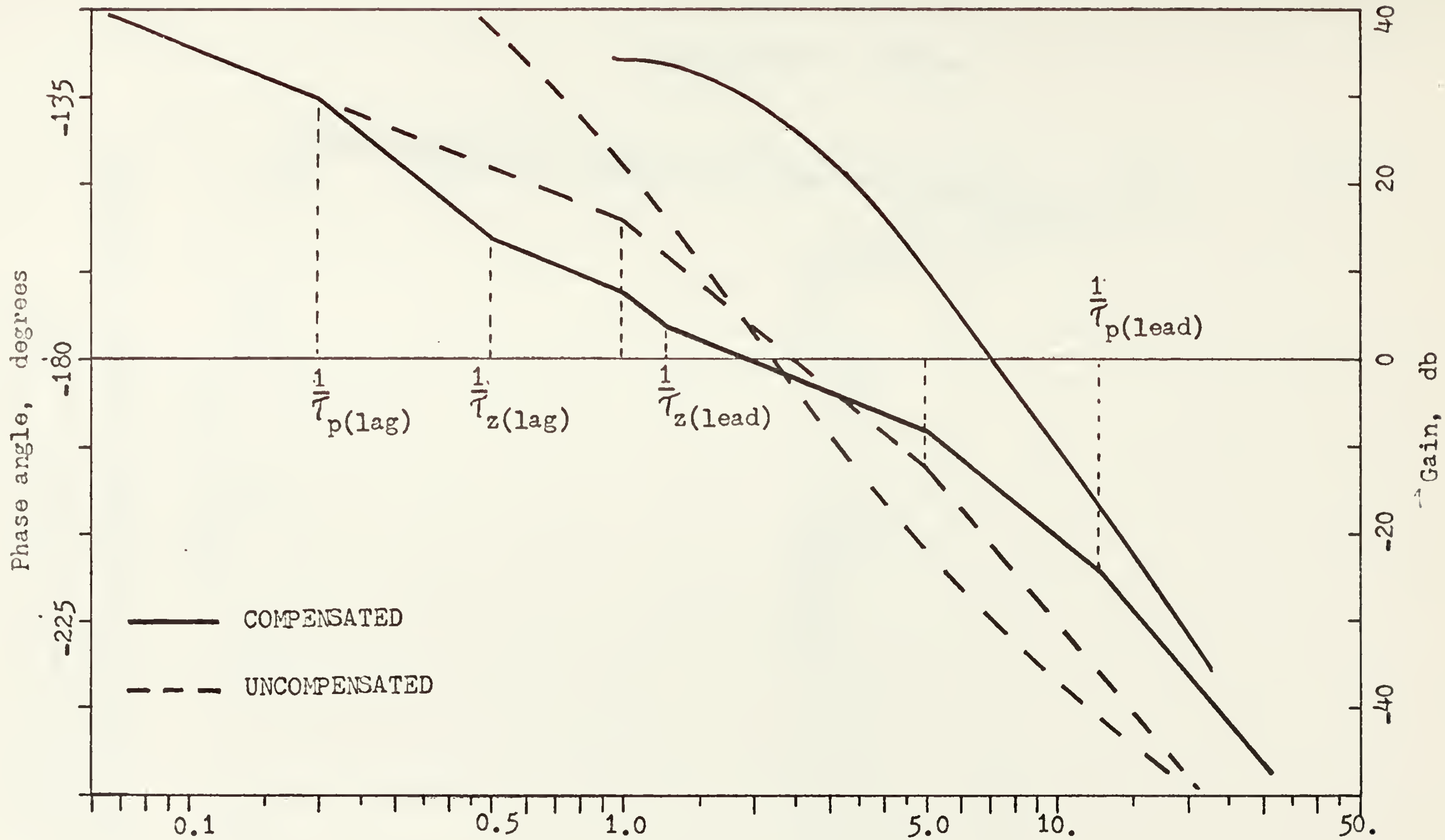



Figure 66. Lag-lead compensated third order servo
$p_{lag}=0.2$, $z_{lag}=0.5$, $p_{lead}=12.5$, $z_{lead}=1.25$
Bode plot

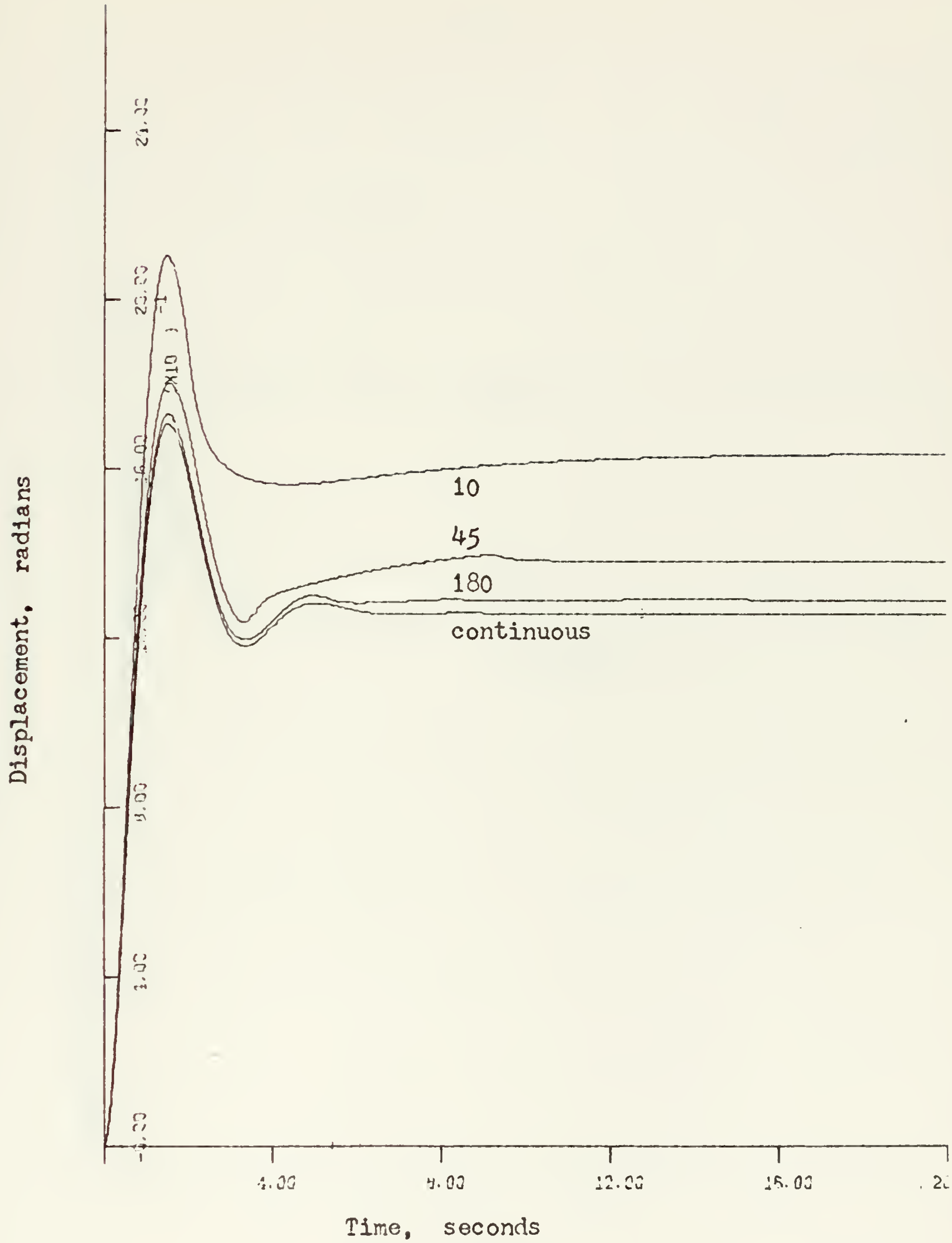



Figure 67. Lag-lead compensated third order servo

$p_{lag}=0.25$, $z_{lag}=0.5$, $p_{lead}=20$, $z_{lead}=2$

OP = 10, 45, 180, and continuous

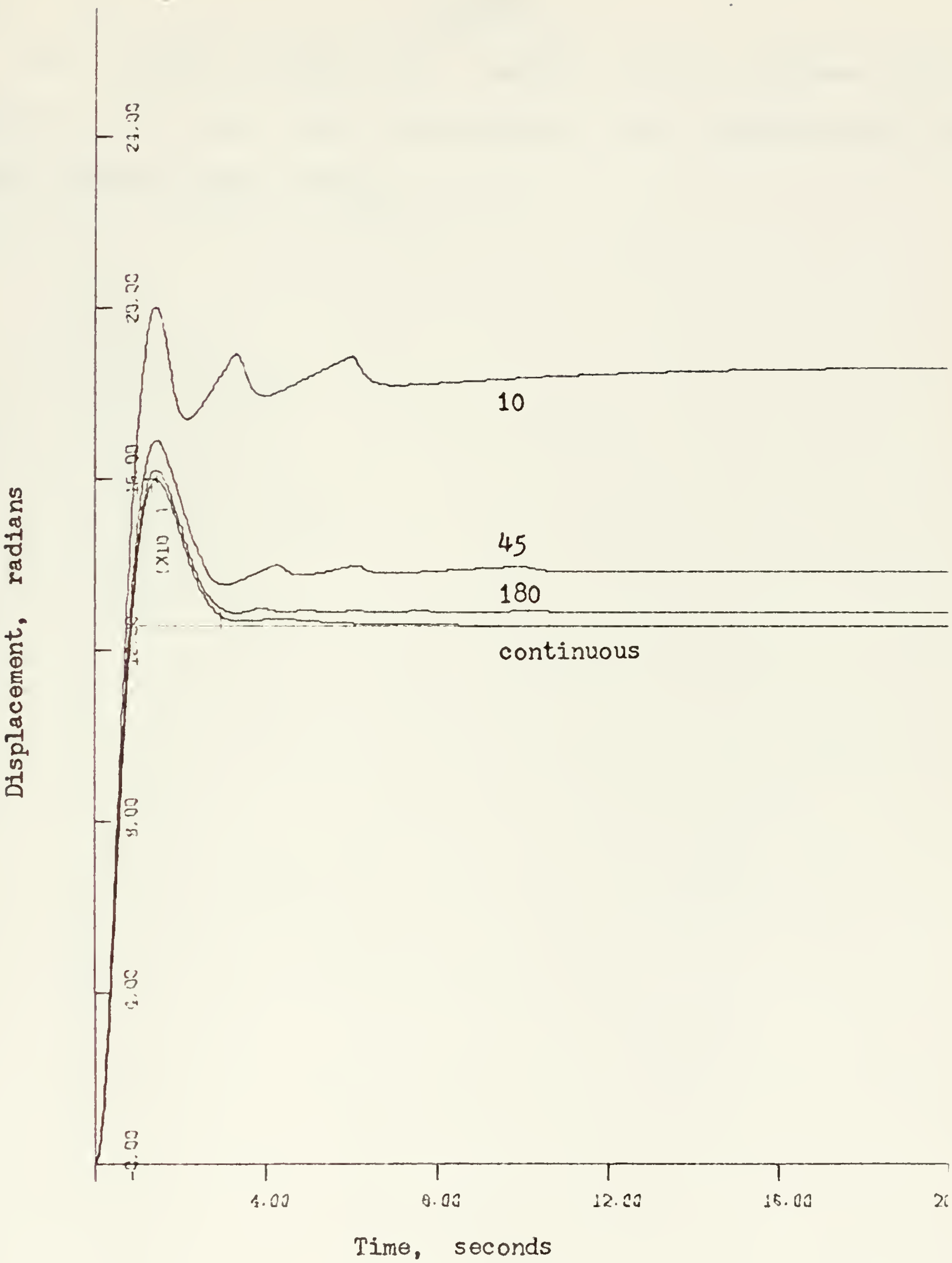



Figure 68. Lag-lead compensated third order servo
$p_{lag}=0.2$, $z_{lag}=0.5$, $p_{lead}=12.5$, $z_{lead}=1.25$
OP = 10, 45, 180, and continuous

# APPENDIX

This appendix presents four examples of the programs used to simulate the system under investigation and a program used to plot the parameter plane curves.

```
TITLE   SECOND ORDER ANALOG POSITIONING SERVO
*       STEP RESPONSE AND PHASE PLANS PLOT
*
*
*
INTEG RKSFX
INTGER NPLOT,NCUR
CONST NPLOT=2, NCUR=2,KV=1.0E7,R=1.25664
INITIAL
DERIVATIVE
      VT=R-C
      CDD=VT*KC
      CDOT=REALPL(0.0,1.0,CDD)
      C=INTGRL(0.0,CDOT)
CONTRL FINTIM=0.01,DELT=0.000001,DELS=0.000002
PRINT 0.0001,VT,CDOT,C
SAMPLE
      CALL DRWG(1,1TIME,C)
      CALL DRWG(2,1,C,CDOT)
TERMINAL
      CALL ENDRW(NPLOT)
END
STOP
```

```
TITLE   SECOND ORDER DIGITAL POSITIONING SERVO
*       STEP RESPONSE AND OUTPUT OF THE UP-DOWN COUNTER
*
*       A IS A FACTOR USED TO ADJUST THE SIZE OF THE CURVE
*
*       MULTIPLE RUNS
*
INTEG RKSFX
INTGER NPLOT,NCUR
CONST NPLOT=1,NCUR=5,OP=1,RN=1,KV=1.0E7,A=20.0
INITIAL
      KC=OP/6.2831853
DERIVATIVE
      BIT=C*KC
      IGBIT=AINT(BIT)
      NT=RN-IGBIT
      VT=NT/KC
      CDD=VT*KV
      CDOT=REALPL(0.0,1.0,CDD)
      C=INTGRL(0.0,CDOT)
      NO=NT*A
CONTRL FINTIM=0.01,DELT=0.000001,DELS=0.00002
PRINT 0.0001,VT,CDOT,C,NT
SAMPLE
      CALL DRWG(NPLOT,1,TIME,NO)
      CALL DRWG(NPLOT,2,TIME,C)
TERMINAL
      IF(NCUR.EQ.1) GO TO 6
      NPLOT=NPLOT+1
      NCUR=NCUR-1
      IF(NCUR.EQ.4) GO TO 1
      IF(NCUR.EQ.3) GO TO 2
      IF(NCUR.EQ.2) GO TO 3
      IF(NCUR.EQ.1) GO TO 4
    1 OP=10.0
      RN=2.0
      A=1.5
      GO TO 5
    2 OP=45.0
      RN=9.0
      A=0.3
      GO TO 5
    3 OP=90.0
      RN=18.0
      A=0.15
      GO TO 5
    4 OP=180.0
      RN=36.0
      A=0.075
    5 CALL RERUN
      RETURN
```

```
      6 CALL ENDRW(NPLOT)
      END
      STOP
```

```
TITLE   THIRD ORDER DIGITAL SERVO
*       VELOCITY AND OUTPUT OF UP-DOWN COUNTER
*       VARIATION OF SLOPE OF RAMP FUNCTION
*
*       MULTIPLE RUNS
*
INTEG SKSFX
INTGER NPLOT,NCUR
CONST NPLOT=1,NCUR=4,KV=2.5,OP=10.0,B=5.0
INITIAL
      KC=OP/6.2831853
DERIVATIVE
      RAM=B*RAMP(0.0)
      RN=AINT(RAM)
      BIT=C*KC
      IGBIT=AINT(BIT)
      NT=RN-IGBIT
      VT=NT/KC
      CD=VT*KV
      CDD=REALPL(0.0,0.2,CD)
      CDOT=REALPL(0.0,1.0,CDD)
      C=INTGRL(0.0,CDOT)
      NO=NT
CONTRL FINTIM=30.0,DELT=0.004,DELS=0.03
PRINT 0.2,VT,CDOT,C,NT
SAMPLE
      CALL DRWG(NPLOT,1,TIME,NO)
      CALL DRWG(NPLOT,2,TIME,CDOT)
TERMINAL
      IF(NCUR.EQ.1) GO TO 6
      NCUR=NCUR-1
      NPLOT=NPLOT+1
      IF(NCUR.EQ.3) GO TO 1
      IF(NCUR.EQ.2) GO TO 2
      IF(NCUR.EQ.1) GO TO 3
    1 B=4.0
      GO TO 5
    2 B=3.0
      GO TO 5
    3 B=2.5
    5 CALL RERUN
      RETURN
    6 CALL ENDRW(NPLOT)
END
STOP
```

```
TITLE  THIRD ORDER POSITIONING SERVO
*      STEP RESPONSE
*
*      MULTIPLE RUNS
*
*      ANALOG AND DIGITAL SYSTEMS
*
INTEG RKSFX
INTGER NPLOT,NCUR
CONST NPLOT=4,NCUR=1,C=0.0,D=1.0,KV=6.0,RN=2.0,OP=10.0,P1=0.2,...
      Z1=0.5,P2=12.5,Z2=1.25
INITIAL
      KC=OP/6.2831853
      TZ1=1.0/Z1
      TP1=1.0/P1
      TZ2=1.0/Z2
      TP2=1.0/P2
DERIVATIVE
      BIT=KC*C*D
      IGBIT=AINT(BIT)
      NT=RN-IGBIT-C*C
      VT=NT*(D/KC+C)
      VIN=VT*KV
      CD=LEDLAG(0.0,TZ1,TP1,VIN)
      COD=LEDLAG(0.0,TZ2,TP2,CD)
      CDD=REALPL(0.0,0.2,COD)
      CDOT=REALPL(0.0,1.0,CDD)
      C=INTGRL(0.0,CDOT)
CONTRL FINTIM=20.0,DELT=0.0025,DELS=0.015
PRINT 0.2,VT,CDOT,C,NT
SAMPLE
      CALL DRWG(1,NCUR,TIME,C)
TERMINAL
      IF(NPLOT.EQ.1) GO TO 6
      NPLOT=NPLOT-1
      NCUR=NCUR+1
      IF(NPLOT.EQ.3) GO TO 2
      IF(NPLOT.EQ.2) GO TO 3
      IF(NPLOT.EQ.1) GO TO 4
    2 RN=9.0
      OP=45.0
      GO TO 5
    3 RN=36.0
      OP=180.0
      GO TO 5
    4 RN=1.25664
      C=1.0
      D=0.0
    5 CALL RERUN
      RETURN
    6 CALL ENDRW(NPLOT)
END
```

```
C****************** PARAMETER PLANE USING MATRIX METHODS ******************* C01
C******************                  PARAM M                ******************* C02
C                                                                          * C03
C     PROGRAM PARAM M CALCULATES AND PLOTS CURVES ON THE TWO DIMENSIONAL   * C04
C     PARAMETER PLANE OF EITHER:                                           * C05
C        1) CONSTANT SIGMA(FNC OMEGA) AND/OR CONSTANT OMEGA(FNC SIGMA);    * C06
C        2) CONSTANT ZETA(FNC OMEGA(N)) AND/OR CONSTANT OMEGA(N)(FNC ZETA); * C07
C        3) SIGMA (REAL ROOT) LINES;                                       * C08
C        4) A COMBINATION OF 1 AND 3 OR 2 AND 3 ABOVE;                     * C09
C     FOR CHARACTERISTIC EQUATIONS OF THE TYPE                             * C10
C        A(0) + A(1)*S + - - - + A(N-2)*S**(N-2) + A(N-1)*S**(N-1) + S**N  * C11
C     WHERE THE A COEFFICIENTS ARE OF THE LINEAR FORM                      * C12
C        B*ALPHA + C*BETA + D  WITH B,C, AND D CONSTANTS.                  * C13
C     IN ADDITION, PROVISIONS ARE INCLUDED IN THE PROGRAM TO COMPUTE THE   * C14
C     ROOTS OF THE REDUCED CHARACTERISTIC EQUATION FOR THE GIVEN SIGMA     * C15
C     AND OMEGA OR ZETA AND OMEGA(N) TO ALLOW FOR SHADING (STABILITY) OF   * C16
C     THE PARAMETER PLANE.                                                 * C17
C
C                                                                          * C18
C************** DEFINITION OF SYMBOLS USED FOR INPUT DATA ***************** C19
C                                                                          * C20
C     NT= TYPE OF OUTPUT CURVES DESIRED                                    * C21
C        1=CONSTANT SIGMA(FNC OMEGA) AND/OR CONSTANT OMEGA(FNC SIGMA)      * C22
C        2=CONSTANT ZETA(FNC OMEGA(N)) AND/OR CONSTANT OMEGA(N)(FNC ZETA)  * C23
C        3=SIGMA(REAL ROOT) LINES                                          * C24
C        4=COMBINATION OF 1 AND 3 ABOVE                                    * C25
C        5=COMBINATION OF 2 AND 3 ABOVE                                    * C26
C           NOTE: ALL CURVES ARE PLOTTED ON PLAIN PAPER.  IF A 1 INCH GRID * C27
C                 IS DESIRED REPLACE ABOVE NT VALUES WITH 11,12,13,14,15   * C28
C     NO= ORDER OF CHARACTERISTIC EQUATION (11>NO>1)                       * C29
C     ND= NO. OF DECADES SPANNED BY OMEGA OR OMEGA(N)IF NT=1,2,4,5         * C30
C     NL= LARGEST SIGMA/ZETA FOR CONSTANT OMEGA/OMEGA(N) CURVES NT=1,2,4,5 * C31
C     NZ= NO. OF CONSTANT: SIGMA CURVES IF NT=1,4; ZETA CURVES IF NT=2,5   * C32
C     NW= NO. OF CONSTANT: OMEGA CURVES IF NT=1,4;OMEGA(N) CURVES IF NT=2,5 * C33
C     NS= NO. OF SIGMA(REAL ROOT) LINES IF NT=3,4,5                        * C34
C     IP= PRINTED OUTPUT DESIRED                                           * C35
C        1=FIRST PLOT PT EACH CURVE PLUS SUB PLOT PTS IF STABILITY CHANGES * C36
C        2=OUTPUT EVERY TENTH PLOTTED POINT PLUS 1 ABOVE                   * C37
C        3=OUTPUT EVERY PLOTTED PT (MAX OF 300+100*(NO-2) LINES PER CURVE) * C38
C        4=OUTPUT EVERY ITERATION STEP (300+100*(NO-2)LINES PER CURVE)     * C39
C        5=OUTPUT EVERY ITERATION WITHOUT ROOTS (300 LINES PER CURVE)      * C40
C        6=OUTPUT EVERY TENTH ITERATION WITHOUT ROOTS (30 LINES PER CURVE) * C41
C        7=NO PRINTED OUTPUT                                               * C42
C     IX= DISTANCE IN INCHES OF THE X-AXIS FROM THE BOTTOM OF THE GRAPH    * C43
C     IY= DISTANCE IN INCHES OF THE Y-AXIS FROM LEFT SIDE OF THE GRAPH     * C44
C     WN= START VALUE OF OMEGA/OMEGA(N) FOR CONSTANT SIGMA/ZETA CURVES     * C45
C     XS= X-SCALE IN UNITS PER INCH (CANNOT BE ZERO)                       * C46
C     YS= Y-SCALE IN UNITS PER INCH (CANNOT BE ZERO)                       * C47
```

```
C     LZ= LABELS FOR CONSTANT: SIGMA(NI=1,4) OR ZETA (NI=2,5) CURVES        * C48
C     LW= LABELS FOR CONSTANT: OMEGA(NI=1,4) OR OMEGA(N)(NI=2,5)CURVES      * C49
C     LS= LABELS FOR SIGMA (REAL ROOT) LINES                                * C50
C                                                                           * C51
C
C************************** DATA CARDS REQUIRED ****************************** C52
C                                                                           * C53
C     CARD 1  FIRST LINE OF GRAPH TITLE (COLUMNS 1-48)                      * C54
C     CARD 2  SECOND LINE OF GRAPH TITLE (COLUMNS 1-48)                     * C55
C     CARD 3  NT,NO,ND,NL,NZ,NW,NS,IP,IX,IY,WN,XS,YS (10I5,3E10.5 FORMAT)   * C56
C     CARD 4  LZ (20A4 FORMAT), USE BLANK CARD IF NI=3                      * C57
C     CARD 5  LW (20A4 FORMAT), USE BLANK CARD IF NI=3                      * C58
C     CARD 6  LS (20A4 FORMAT), USE BLANK CARD IF NI=1,2                    * C59
C     CARD 7  CORRESPONDING VALUES OF CARD 4 (8E10.5 FORMAT)                * C60
C     CARD 8  CORRESPONDING VALUES OF CARD 5 (8E10.5 FORMAT)                * C61
C     CARD 9  CORRESPONDING VALUES OF CARD 6 (8E10.5 FORMAT)                * C62
C     CARD 10 CONSTANT COEFFICIENTS IN ASCENDING ORDER (8E10.5 FORMAT)      * C63
C     CARD 11 ALPHA COEFFICIENTS IN ASCENDING ORDER (8E10.5 FORMAT)         * C64
C     CARD 12 BETA  COEFFICIENTS IN ASCENDING ORDER (8E10.5 FORMAT)         * C65
C     CARD 13 ALPHA-BETA PROD COEFF IN ASCENDING ORDER (8E10.5 FORMAT)      * C66
C                                                                           *
C****************************************************************************
C************************** SAMPLE DATA DECK ******************************** C67
C                              (SCALE COMPRESSED)                           * C68
C                                                                           * C69
C   LINE |   |  |       |       |       |       |       |       |       |   * C70
C        1   5  10      20      30      40      50      60      70      80  * C71
C                                                                           * C72
CARD 1   DUBAC  PARAMETER PLANE CURVES                                      * C73
CARD 2   S4+(20+A)S3+(1700+20A)S2+(1700A+B)S+25B                            * C74
CARD 3      2   4   2   1   6   5   0   1   0   0  1.0      3.0   .3  E+04  * C75
CARD 4   Z0.0Z0.2Z0.4Z0.6Z0.8Z1.0                                           * C76
CARD 5   WN 5WN10WN20WN30WN40                                               * C77
CARD 6   (BLANK CARD)                                                       * C78
CARD 7      0.0       0.2       0.4       0.6       0.8       1.0           * C79
CARD 8      5.0      10.0      20.0      30.0      40.0                     * C80
CARD 9                                                                      * C81
CARD 10     0.0       0.0    .17 E+04  20.0                                 * C82
CARD 11     0.0    .17 E+04  20.0       1.0                                 * C83
CARD 12    25.0       1.0                                                   * C84
CARD 13                                                                     * C85
C
C
C******************** LIMITATIONS AND ERROR READOUTS ************************ C86
C                                                                           * C87
C     LIMITATIONS:                                                          * C88
C        1. ORDER OF CHARACTERISTIC EQUATION LIMITED TO 10.  CAN BE         * C89
C           INCREASED TO N BY CHANGING DIMENSIONS OF AK TO N,N; CK,BK,DK,   * C90
```

```
C           SIGP TO N;  COEF,COF TO N-1;  ROOTR,ROOTI TO N-2; AND AA TO NXN * C91
C        2. NUMBER OF CURVES LIMITED TO 20 PER TYPE (MAXIMUM OF SIXTY).     * C92
C           CAN BE INCRESED TO M BY CHANGING DIMENSIONS OF LZ,LW,LS,SZ,WWN, * C93
C           SIG TO M.                                                       * C94
C        3. CAUTION** IF ALL ALPHA COEF. AND OR ALL BETA COEF EQUAL ZERO -  *AC94
C           YOU DO NOT HAVE A VALID NONLINEAR TWO PARAMETER PROBLEM.        *BC94
C           CHECK TO SEE IF YOU CAN REFORMULATE THE PROBLEM (I.E. SET       *CC94
C           BETA COEF = ALPHA-BETA COEF)                                    *DC94
C                                                                           * C95
C     ERROR READOUTS:                                                       * C96
C        1. POSSIBLE SINGULAR POINT                                         * C97
C              MEANING: MATRIX BEING INVERTED EITHER SINGULAR OR NEARLY SO. * C98
C        2. POLRT ERROR. COEF OF REDUCED CHAR. EQN.:                        * C99
C              MEANING: SUBROUTINE POLRT UNABLE TO DETERMINE ROOTS WITH 500 *C100
C                       ITERATIONS ON 5 STARTING VALUES.  COEFFICIENTS ARE  *C101
C                       PRINTED OUT FOR CONVENIENCE.                        *C102
C                                                                           *C103
C************************* TEMPORARY RESTRICTION **************************C104
C                                                                           *C105
C     DUE TO SOME AS YET UNDETERMINED QUIRK, AN ERROR IN THE POLRT SUB-     *C106
C     ROUTINE PERIODICALLY CAUSES AN ERROR INTERRUPT WHICH IN THE MVT MODE  *C107
C     OF OPERATION ON THE IBM 360 SELF CORRECTS ITSELF.  UNTIL SUCH TIME    *C108
C     AS THIS DESCREPANCY CAN BE ELIMINATED, THIS PROGRAM MUST BE RUN ON MVT*C109
C     ONLY.  IF ANY SUCH INTERRUPTS OCCUR, THE FOLLOWING MESSAGE WILL       *C110
C     APPEAR AT THE END OF THE PRINT OUT:                                   *C111
C           SUMMARY OF ERRORS FOR THIS JOB  ERROR NUMBER  NUMBER OF ERRORS  *C112
C                                               209         "NO. ERRORS"    *C113
C**************************************************************************C114
      REAL*8 ITITLE(12),AK(10,12),AA(10,12),X,Y,Z                           0002
      REAL LABEL/4H    /,GG(10)/1.0076,1.016,1.0245,1.0312,1.0394,          0003
     11.0483,1.0568,1.0633,1.071,1.078/,LZ(20),                             0004
     2LW(20),LS(20),SZ(20),WWN(20),SIG(20),CK(20),BK(10),DK(10),            005
     *COEF(9),COF(9),ROOTR(8),ROOTI(8),ALFA(300),BETA(300),SIGP(10)         0006
      CALL ERRSET (207,256,-1,1,0,209)                                      TEMP
      IPROD = 1                                                             00A6
      WRITE(6,500)                                                          0007
      READ(5,400)(ITITLE(I),I=1,12)                                         0008
      WRITE(6,401)(ITITLE(I),I=1,12)                                        0009
      READ(5,402) NT,NO,ND,NL,NZ,NW,NS,IP,IX,IY,WN,XS,YS                    0010
      WRITE(6,501)                                                          0011
      WRITE(6,411) NT,NO,ND,NL,NZ,NW,NS,IP,IX,IY,WN,XS,YS                   0012
C
C     DETERMINE IF 1 INCH GRID IS DESIRED IF SO IG = 1
      IG=0                                                                  0013
      IF(NT.LE.5) GO TO 200                                                 0014
      IG=1                                                                  0015
      NT=NT-10                                                              0016
  200 M1=NO+1                                                               0017
```

```
      M2=NO-2
      M3=NO-1
      M4 = NO + 2
C
C     READ AND WRITE CONSTANT SIGMA(NT=1,4) OR ZETA(NT=2,5) LABELS
      READ(5,403)(LZ(I),I=1,NZ)
      WRITE(6,502)
      WRITE(6,404)(LZ(I),I=1,NZ)
C
C     READ AND WRITE CONSTANT OMEGA(NT=1,4) OR OMEGA(N)(NT=2,5) LABELS
      READ(5,403)(LW(I),I=1,NW)
      WRITE(6,503)
      WRITE(6,404)(LW(I),I=1,NW)
C
C     READ AND WRITE CONSTANT SIGMA(NT=3) LABELS   (FOR REAL ROOTS)
      READ(5,403)(LS(I),I=1,NS)
      WRITE(6,504)
      WRITE(6,404)(LS(I),I=1,NS)
C
C     READ AND WRITE CONSTANT SIGMA(NT=1,4) OR ZETA(NT=2,5) VALUES
      READ(5,405)(SZ(I),I=1,NZ)
      WRITE(6,505)
      WRITE(6,406)(SZ(I),I=1,NZ)
C
C     READ AND WRITE CONSTANT OMEGA(NT=1,4) OR OMEGA(N)(NT=2,5) VALUES
      READ(5,405)(WWN(I),I=1,NW)
      WRITE(6,506)
      WRITE(6,406)(WWN(I),I=1,NW)
C
C     READ AND WRITE CONSTANT SIGMA(NT=3) VALUES   (FOR REAL ROOTS)
      READ(5,405)(SIG(I),I=1,NS)
      WRITE(6,507)
      WRITE(6,406)(SIG(I),I=1,NS)
C
C     READ AND WRITE COEFFICIENT CONSTANTS
      READ(5,405)(CK(M1-I),I=1,NO)
      WRITE(6,508)
      WRITE(6,406)(CK(M1-I),I=1,NO)
C
C     READ AND WRITE ALPHA COEFFICIENTS
      READ(5,405) (AK(M1-I,1),I=1,NO)
      WRITE(6,509)
      WRITE(6,406)(AK(M1-I,1),I=1,NO)
C
C     READ AND WRITE BETA COEFFICIENTS
      READ(5,405) (AK(M1-I,2),I=1,NO)
      WRITE(6,510)
      WRITE(6,406)(AK(M1-I,2),I=1,NO)
```

```
      READ(5,405)(AK(M1-I,M4),I=1,NO)
C
C     DETERMINE IF ALPHA-BETA PROD COEFF ARE NON-ZERO
      DO 600 I=1,NO
      IF(AK(I,M4).NE.0.) GO TO 601
  600 CONTINUE
      WRITE(6,610)
  610 FORMAT(///,T3,'ALPHA-BETA COEFFICIENTS ARE ZERO - LINEAR CASE')
      GO TO 602
  601 IPROD = 2
      WRITE(6,527)
  527 FORMAT(///,T3,'CONSTANT ALPHA-BETA PRODUCT COEFFICIENTS IN ASCEN',
     1'DING ORDER',/)
      WRITE(6,406) (AK(M1-I,M4),I=1,NO)
  602 MCD=1
C
C     SET GRAPH INDICES
      XLIMP=(9.2-IY)*XS
      YLIMP=(15.2-IX)*YS
      XLIMN=-IY*XS
      YLIMN=-IX*YS
      IF(NO.EQ.2) GO TO 202
C
C     INITIALIZE NON-COEFFICIENT PORTION OF AK MATRIX
      DO 101 I=1,NO
      DO 100 J=1,M3
  100 AK(I,J+2)=0.0
  101 BK(I)=0.0
C
C     SET SECOND SUPER-DIAGONAL OF AK MATRIX TO -1.
      DO 102 I=1,M2
  102 AK(I,I+2)=-1.0
  202 GO TO(103,103,147,103,103),NT
  103 DO 146 JJ=1,2
C
C     JJ=1 DETERMINE CONSTANT SIGMA(NT=1,4),ZETA(NT=2,5) CURVES
C     JJ=2 DETERMINE CONSTANT OMEGA(NT=1,4),OMEGAN(NT=2,5) CURVES
      GO TO (204,205),JJ
  204 IF(NZ.EQ.0) GO TO 146
C
C     SET NI=NUMBER CONSTANT SIGMA(NT=1,4),ZETA(NT=2,5) CURVES
      NI=NZ
C
C     G = INCREMENT FOR OMEGA(OMEGAN) TO SPAN REQUIRED NUMBER OF DECADES
      G=GG(ND)
      WRITE(6,526)
      GO TO 206
  205 IF(NW.EQ.0) GO TO 146
```

```
C       SET NI=NUMBER CONSTANT OMEGA(NT=1,4),OMEGAN(NT=2,5) CURVES
        NI=NW                                                          0068
C       DEL= INCREMENT FOR SIGMA(ZETA) TO SPAN DESIRED RANGE
        DEL=NL/300.0                                                   0069
        WRITE(6,526)                                                   0070
    206 DO 145 K=1,NI                                                  0071
        IQUAD = 1                                                      0A71
        DO 145 IDL=1,IPROD                                             0B71
        IF(IQUAD.EQ.0) GO TO 145                                       0C71
        SN = 1.                                                        0D71
        IF(IQUAD.EQ.2) SN = -1.                                        0E71
        GO TO (104,207),JJ                                             0072
C
C       SET INITIAL VALUE OF OMEGA(OMEGAN) FOR CONSTANT SIGMA(ZETA) CURVE
    104 W=WN                                                           0073
C
C       SIGZ = PRESENT CONSTANT SIGMA(ZETA) VALUE
        SIGZ=SZ(K)                                                     0074
        GC TO 105                                                      0075
C
C       SET INITIAL VALUE OF SIGMA(ZETA) FOR CONSTANT OMEGA(OMEGAN) CURVE
    207 SIGZ=0.0                                                       0076
C
C       WWN = PRESENT CONSTANT OMEGA(OMEGAN) VALUE
        W=WWN(K)                                                       0077
C
C       LL = FLAG WHICH COUNTS THE NUMBER OF POINTS ON THE PRESENT CURVE
    105 LL=0                                                           0078
        LLL=0                                                          0079
        LLLL=0                                                         0080
        IPSET=0                                                        0081
        IQUAD = 0                                                      0A81
        DO 242 L=1,300                                                 0082
        IPP=0                                                          0083
        GO TO(106,109,146,106,109),NT                                  0084
    106 IF(L.GT.1) GO TO 607                                           0085
        GO TO (306,307),JJ                                             0086
    306 WRITE(6,511)                                                   0087
        WRITE(6,513)                                                   0088
        GO TO 607                                                      0089
    307 WRITE(6,512)                                                   0090
        WRITE(6,513)                                                   0091
    607 IF(NO.EQ.2) GO TO 208                                          0092
C
C       SET SUPER DIAGONAL OF AK MATRIX = -2*SIGMA (NT=1,4N
        DO 107 I=2,M3
    107 AK(I,I+1)=-2.0*SIGZ                                            0094
C       SET DIAGONAL OF AK MATRIX = -(SIGMA**2 + OMEGA**2) (NT=1,4)
```

```
      DO 108  I=3,NO                                                    0095
  108 AK(I,I)=-(SIGZ**2  + W**2)                                        0096
  208 BK(1)=2.0*SIGZ                                                    0097
      BK(2)=SIGZ**2+W**2                                                0098
      GO  TO  112                                                       0099
  109 IF(L.GT.1)  GO  TO  310                                           0100
      GO  TO  (209,210),JJ                                              0101
  209 WRITE(6,522)                                                      0102
      WRITE(6,524)                                                      0103
      GO  TO  310                                                       0104
  210 WRITE(6,523)                                                      0105
      WRITE(6,524)                                                      0106
  310 IF(NO.EQ.2)  GO  TO  211                                          0107
C
C
C     SET  SUPER  DIAGONAL  OF  AK  MATRIX  =  -2*ZETA*OMEGA(N)  (NT=2,5)
      DO  110  I=2,M3                                                   0108
  110 AK(I,I+1)=-2.0*SIGZ*W                                             0109
C
C     SET  DIAGONAL  OF  AK  MATRIX  =  -OMEGA(N)**2  (NT=2,5)
      DO  111  I=3,NO                                                   0110
  111 AK(I,I)=-W**2                                                     0111
  211 BK(1)=2.0*SIGZ*W                                                  0112
      BK(2)=W**2                                                        0113
  112 DO  113  I=1,NO                                                   0114
  113 AK(I,M1)  =  CK(I)  -  BK(I)                                      115-6
      DO  115  I=1,NO                                                   0117
      DO  114  J=1,M4                                                   0118
  114 AA(I,J)  =  AK(I,J)                                              119-20
  115 CONTINUE                                                          0121
      CALL  SOLVE(AA,NO,NO+IPROD,KS)                                    0122
      GO  TO(603,116),KS                                                123-4
  116 WRITE(6,407)SIGZ,W                                                0125
      GO  TO  240                                                       0126
  603 GO  TO(604,605),IPROD                                             A126
  604 DO  620  I=1,NO                                                   B126
  620 DK(I)  =  -AA(I,M1)                                               C126
      GO  TO  117                                                       D126
  605 IF(AA(1,M4))606,637,606                                           E126
  637 DK(1)  =  -AA(1,M1)                                               F126
      DK(2)  =  -AA(2,M1)/(1.-AA(2,M4)*AA(1,M1))                        G126
      GO  TO  615                                                       H126
  606 IF(AA(2,M4))  608,609,608                                         I126
  609 DK(1)  =  -AA(1,M1)/(1.  -  AA(1,M4)*AA(2,M1))                    J126
      DK(2)  =  -AA(2,M1)                                               K126
      GO  TO  615                                                       L126
  608 X  =  (-1.  +  AA(1,M4)*AA(2,M1)  -  AA(1,M1)*AA(2,M4))/AA(2,M4)  M126
      Y  =  AA(1,M1)/AA(2,M4)                                           N126
```

```
      Z = X**2 - 4.*Y                                                   0126
      IF(Z) 240,611,612                                                 P126
  611 DK(1) = .5*X                                                      Q126
      GO TO 613                                                         R126
  612 IQUAD = 2                                                         S126
      Z = SN*DSQRT(Z)                                                    T126
      DK(1) = (X + Z)/2.                                                U126
  613 DK(2) = -AA(2,M1)/(1. + AA(2,M4)*DK(1))                           V126
      IF(NO.EQ.2) GO TO 117                                             W126
  615 DO 614 I=3,NO                                                     X126
  614 DK(I) =  -AA(I,M1)-AA(I,M1)*DK(1)*DK(2)                           Y126
  117 ATEMP=DK(1)                                                       0127
      BTEMP=DK(2)                                                       0128
      IF(L.GT.1) GO TO 217                                              0129
      AMAX=ATEMP                                                        0130
      AMIN=ATEMP                                                        0131
      BMAX=BTEMP                                                        0132
      BMIN=BTEMP                                                        0133
      GO TO 717
C
C     DETERMINE RUNNING MAXIMUM AND MINIMUM OF ALPHA AND BETA
  217 IF(ATEMP.GE.AMIN) GO TO 317                                       0134
      AMIN=ATEMP                                                        0135
  317 IF(ATEMP.LE.AMAX) GO TO 417                                       0136
      AMAX=ATEMP                                                        0137
  417 IF(BTEMP.GE.BMIN) GO TO 617                                       0138
      BMIN=BTEMP                                                        0139
  617 IF(BTEMP.LE.BMAX) GO TO 717                                       0140
      BMAX=BTEMP                                                        0141
  717 IF(NO.EQ.2) GO TO 218                                             0142
      DO 118 I=1,M2                                                     0143
  118 COEF(I)=DK(M1-I)                                                  0144
      COEF(M3)=1.0                                                      0145
C
C     DETERMINE IF PRESENT ALPHA OR BETA EXCEEDS GRAPH LIMITS
  218 IF(XLIMN-ATEMP)119,119,123                                        0146
  119 IF(ATEMP-XLIMP)120,120,123                                        0147
  120 IF(YLIMN-BTEMP)121,121,123                                        0148
  121 IF(BTEMP-YLIMP) 122,122,123                                       0149
  122 LL=LL+1                                                           0150
      LLL=LLL+1                                                         0151
      GO TO (125,125,125,125,138,234,140),IP                            0152
  123 GO TO (240,240,240,124,124,124,240),IP                            0153
  124 IPP=1                                                             0154
      GO TO (125,125,125,125,138,234,142),IP                            0155
  125 IF(NO.GT.3) GO TO 225                                             0156
      IF(NO.EQ.2) GO TO 127                                             0157
      RCOTR(1)=-DK(3)                                                   0158
```

```
      ROOTI(1)=0.0
      GO TO 127
C
C     OBTAIN ROOTS OF REMAINING POLYNOMIAL
  225 CALL POLRT(COEF,COF,M2,ROOTR,ROOTI,IER)
      IERROR=IER+1
      GO TO(127,126,126,126,126),IERROR
  126 WRITE(6,408)ATEMP,BTEMP,SIGZ,W
      WRITE(6,412) (COEF(M3+1-I),I=1,M3)
      IF(LL.GT.1) GO TO 139
      IPSET=1
      GO TO 139
  127 GO TO (128,128,138,138,138,138,142),IP
  128 ISTAB=0
      IF(NO.EQ.2) GO TO 130
C
C     CHECK FOR CHANGE IN STABILITY IN REMAINING POLYNOMIAL
      DO 130  J=1,M2
      IF(ROOTR(I)) 130,130,129
  129 ISTAB=1
  130 CONTINUE
      IF(LL.EQ.1.OR.IPSET.EQ.1) GO TO 231
      GO TO 131
  231 ISTABL=ISTAB
  131 IF(ISTABL-ISTAB)335,132,335
  132 IF(LL.EQ.1.OR.IPSET.EQ.1) GO TO 138
      GO TO (140,133,138,138,138,138,142),IP
  133 IF(LLL-10)140,134,134
  134 LLL=0
      GO TO 138
  234 IF(L.EQ.1.OR.LLLL.EQ.10) GO TO 235
      GO TO 139
  235 LLLL=0
      GO TO 138
  335 ISTABL=ISTAB
  135 GO TO (138,136,138,138,138,138,240),IP
  136 IF(LLL-10)138,137,137
  137 LLL=0
  138 WRITE(6,409)ATEMP,BTEMP,SIGZ,W
      IPSET=0
      GO TO (238,238,238,238,139,139,240),IP
  238 IF(NO.EQ.2) GO TO 139
      WRITE(6,410)(ROOTR(I),ROOTI(I),I=1,M2)
  139 IF(IPP.EQ.1)GO TO 240
  140 ALFA(LL)=ATEMP
      BETA(LL)=BTEMP
  240 IF(JJ.EQ.1)GO TO 141
      SIGZ=SIGZ+DEL
```

```
      GO TO 142                                                          0203
  141 W=W*G                                                              0204
  142 LLLL=LLLL+1                                                        0205
  242 CONTINUE                                                           0206
      IF(LL.LT.2)GO TO 144                                               0207
      IF(JJ.EQ.1)GO TO 143                                               0208
C
C     PLOT COSTANT OMEGA(OMEGAN) ARRAYS
      CALL DRAW(LL,ALFA,BETA,MOD,0,LW( K),ITITLE,XS,YS,IX,IY,2,2,        0209
     *9,15,IG,LAST)                                                      0210
      WRITE(6,519) W,LL                                                  0211
      WRITE(6,525) AMAX,BMAX,AMIN,BMIN                                   0212
      MOD=2                                                              0213
      GO TO 145                                                          0214
C
C     PLOT CONSTANT SIGMA(ZETA ) ARRAYS
  143 CALL DRAW(LL,ALFA,BETA,MOD,0,LZ(K),ITITLE,XS,YS,IX,IY,2,2,         0215
     *9,15,IG,LAST)                                                      0216
      WRITE(6,520) SIGZ,LL                                               0217
      WRITE(6,525) AMAX,BMAX,AMIN,BMIN                                   0218
      MOD=2                                                              0219
      GO TO 145                                                          0220
  144 GO TO (244,344),JJ                                                 0221
  244 WRITE(6,514) SIGZ                                                  0222
      WRITE(6,525) AMAX,BMAX,AMIN,BMIN                                   0223
      GO TO 145                                                          0224
  344 WRITE(6,521) W                                                     0225
      WRITE(6,525) AMAX,BMAX,AMIN,BMIN                                   0226
  145 CONTINUE                                                           0227
  146 CONTINUE                                                           0228
  147 GO TO (165,165,148,148,148),NT                                     0229
  148 GO TO (248,248,248,248,248,248,249),IP                             0230
  248 WRITE(6,515)                                                       0231
C
C
C     THIS SECTION CALCULATES REAL ROOT LINES
      WRITE(6,680)                                                       0232
  249 DEL = (YLIMP - YLIMN)/300.                                         A232
      DO 164 K=1,NS                                                      B232
      LL = 1                                                             C232
      LLL = 0                                                            D232
C
C     OBTAIN POWERS OF SIGMA
      DO 149 I=1,NO                                                      0233
  149 SIGP(I)=SIG(K)**I                                                  0234
C
C     OBTAIN COEFFICIENTS OF REAL ROOT MONOMIAL
      ACOEF=AK(NO,1)                                                     0235
```

```
      BCOEF=AK(NO,2)
      CCOEF=SIGP(NO)+CK(NO)
      ABCOEF = AK(NO,M4)
      DO 150 I=1,M3
      NOMI = NO - I
      ACOEF = ACOEF + AK(I,1)*SIGP(NOMI)
      BCOEF = BCOEF + AK(I,2)*SIGP(NOMI)
      ABCOEF = ABCOEF + AK(I,M4)*SIGP(NOMI)
  150 CCOEF = CCOEF + CK(I)*SIGP(NOMI)
      TRIAL = YLIMN
      DO 160 JJ=1,300
      DENOM = ACOEF + ABCOEF*TRIAL
      IF(ABS(DENOM).LE.1.E-14) GO TO 153
      ALFA(LL) = -(CCOEF + BCOEF*TRIAL)/DENOM
      IF(XLIMN - ALFA(LL)) 151,151,153
  151 IF(ALFA(LL) - XLIMP) 154,154,153
  154 LLL = LLL + 1
      BETA(LL) = TRIAL
      GO TO (155,155,155,155,155,155,152),IP
  155 IF(LLL - 10) 152,250,250
  250 LLL = 0
      SLOPE = -BCOEF/DENOM
      AINTER = -CCOEF/DENOM
      IF(IPROD.EQ.2) SLOPE = -(CCOEF+BCOEF)/DENOM
      WRITE(6,409) SIG(K),ALFA(LL),BETA(LL),SLOPE,AINTER
  152 LL = LL + 1
  153 TRIAL = TRIAL + DEL
  160 CONTINUE
      LL = LL - 1
      IF(LL.LT.2) GO TO 163
C
C     PLOT REAL ROOT CURVE
  162 CALL DRAW(LL,ALFA,BETA,MOD,0,LS(K),ITITLE,XS,YS,IX,IY,2,2,9,15,0,
     1LAST)
      MOD=2
      GO TO 164
C
C     IF HAVE LESS THAN TWO POINTS  - PRESENT SIGMA LIES OUTSIDE RANGE
  163 WRITE(6,517) SIG(K)
  164 CONTINUE
C
C     IF MOD=1 NO GRAPHS HAVE BEEN PLOTTED
  165 IF(MOD.EQ.1) GO TO 166
C
C     IF MOD=2 PLOT REAL AXIS--FINISH GRAPH
      ALFA(1)=0.0
      ALFA(2)=1.0
      BETA(1)=0.0
```

```
      BETA(2)=0.0
      CALL DRAW(2,ALFA,BETA,3,0,LABEL,ITITLE,XS,YS,IX,IY,2,2,
     *9,15,0,LAST)
      GO TO 167
  166 WRITE(6,518)
  167 CONTINUE
  400 FORMAT(6A8)
  401 FORMAT(2X,6A8)
  402 FORMAT(10I5,3E10.5)
  403 FORMAT(20A4)
  404 FORMAT(2X,20(A4,2X))
  405 FORMAT(8E10.5)
  406 FORMAT( 2X,8E14.5)
  407 FORMAT(T7,'POSSIBLE SINGULAR POINT',2E14.5,T64,'NO ROOTS;',
     *T73,'POINT NOT PLOTTED ON GRAPH')
  408 FORMAT(4E14.5,T73,'POLRT ERROR. COEF OF REDUCED CHAR. EQN.:')
  409 FORMAT(5E14.5)
  410 FORMAT(58X,6E12.4)
  411 FORMAT(10I5,1X,3E12.4)
  412 FORMAT( 76X,4E12.4)
  500 FORMAT('1',T6,'THE INPUT DATA IS:',/////)
  501 FORMAT(//,3X,2HNT,3X,2HNO,3X,2HND,3X,2HNL,3X,2HNZ,3X,2HNW,3X,
     *2HNS,3X,2HIP,3X,2HIX,3X,2HIY,T53,'START OMEGA',T68,'X-SCALE',
     *T80,'Y-SCALE',/)
  502 FORMAT(///,T3,'LABELS FOR CONSTANT SIGMA OR ZETA CURVES',/)
  503 FORMAT(///,T3,'LABELS FOR CONSTANT OMEGA OR OMEGA(N) CURVES',/)
  504 FORMAT(///,T3,'LABELS FOR SIGMA(REAL ROOT) LINES',/)
  505 FORMAT(///,T3,'VALUES FOR CONSTANT SIGMA OR ZETA CURVES',/)
  506 FORMAT(///,T3,'VALUES FOR CONSTANT OMEGA OR OMEGA(N) CURVES',/)
  507 FORMAT(///,T3,'VALUES FOR SIGMA(REAL ROOT) LINES',/)
  508 FORMAT(///,T3,'CONSTANT COEFFICIENTS IN ASCENDING ORDER',/)
  509 FORMAT(///,T3,'CONSTANT ALPHA COEFFICIENTS IN ASCENDING ORDER',/)
  510 FORMAT(///,T3,'CONSTANT BETA  COEFFICIENTS IN ASCENDING ORDER',/)
  511 FORMAT(///,T19,'CONSTANT SIGMA CURVE ',T81,'CORRESPONDING REMAININ
     *G ROOTS',/)
  512 FORMAT(///,T19,'CONSTANT OMEGA CURVE ',T81,'CORRESPONDING REMAININ
     *G ROOTS',/)
  513 FORMAT(T7,'ALPHA',T21,'BETA',T34,'SIGMA',T49,'OMEGA',T64,'REAL',
     *T76,'IMAG',T88,'REAL',T100,'IMAG',T112,'REAL',T124,'IMAG',//)
  514 FORMAT(T8,'CURVE FOR CONSTANT SIGMA/ZETA =',E13.5,T53,'WAS NOT PLO
     *TTED',/,T8,'AS LESS THAN TWO POINTS WERE GENERATED WITHIN SPECIFIE
     *D GRAPH RANGE',/)
  515 FORMAT('1',//,T25,'SIGMA (REAL ROOT) LINES',//)
  516 FORMAT(T3,'FOR SIGMA =',E12.5,T27,': ALPHA = ',E12.5,T50,
     *'* BETA + ',E12.5,/)
  517 FORMAT(T3,'REAL ROOT LINE FOR SIGMA =',E13.5,T42,'LIES OUTSIDE THE
     * SPECIFIED GRAPH RANGE',/)
  518 FORMAT(////,T3,'THERE WILL BE NO OUTPUT GRAPH PLOTTED AS NONE',/,
```

```
      *T3,'OF THE POINTS GENERATED BY THE PROGRAM LIE',/,                0327
      *T3,'WITHIN THE SPECIFIED GRAPH REGION. CHECK THE VALUES OF',/,    0328
      *T3,'XS AND YS AS WELL AS IX AND IY ON DATA CARD 3',/)             0329
  519 FORMAT(T8,'FOR THE CURVE OF CONSTANT OMEGA/OMEGA(N) =',E13.5,      0330
      */,I10,T12,                                                        0331
      *'POINTS OUT OF A POSSIBLE 300 LIE WITHIN SPECIFIED GRAPH RANGE',/) 0332
  520 FORMAT(T8,'FOR THE CURVE OF CONSTANT SIGMA/ZETA =',E13.5,          0333
      */,I10,T12,                                                        0334
      *'POINTS OUT OF A POSSIBLE 300 LIE WITHIN SPECIFIED GRAPH RANGE',/) 0335
  521 FORMAT(T8,'CURVE FOR CONSTANT OMEGA/OMEGA(N) =',E13.5,T57,'WAS NOT 0336
      * PLOTTED',/,T8,'AS LESS THAN TWO POINTS WERE GENERATED WITHIN SPEC 0337
      *IFIED GRAPH RANGE',/)                                             0338
  522 FORMAT(///,T19,'CONSTANT ZETA CURVE ',T81,'CORRESPONDING REMAINING 0339
      * ROOTS',/)                                                        0340
  523 FORMAT(///,T17,'CONSTANT OMEGA(N) CURVE ',T81,'CORRESPONDING REMAI 0341
      *NING ROOTS',/)                                                    0342
  524 FORMAT(T7,'ALPHA',T21,'BETA',T35,'ZETA',T48,'OMEGA(N)',T64,'REAL', 0343
      *T76,'IMAG',T88,'REAL',T100,'IMAG',T112,'REAL',T124,'IMAG',//)     0345
  525 FORMAT(T8,'MAXIMUM AND MINIMUM GENERATED ALPHA & BETA VALUES WERE: 0346
      *',/,T8,'AMAX=',E10.3,T24,'BMAX=',E10.3,T40,'AMIN=',E10.3,T56,     0347
      *'BMIN=',E10.3,/)                                                  0348
  526 FORMAT('1')                                                        0349
  680 FORMAT(//,T7,'SIGMA',T21,'ALPHA',T35,'BETA',T48,'SLOPE',T62,       0350
      ''AINTER')                                                         0351
       END                                                               0352
       SUBROUTINE SOLVE(A,M,N,KER)                                       S1
       REAL*8 A(10,12),TEMPA,BIG,G,F                                     S2
C      MATRIX OF ORDER MXM - AUGMENTED TO N COLUMNS -- IS PUT INTO       S3
C      ECHELON FORM BY ELEMENTARY TRANSFORMATIONS                        S4
       IC = 1                                                            S5
C      SEARCH IC COLUMN ON OR BELOW MAIN DIAGONAL FOR LARGEST ELEMENT    S6
       DO 19 IR =1,M                                                     S7
   11  BIG = DABS(A(IR,IC))                                              S8
       I=IR                                                              S9
       DO 12 K=IR,M                                                      S10
       IF(BIG.GE.DABS(A(K,IC))) GO TO 12                                 S1
       BIG = DABS(A(K,IC))                                               S12
       I = K                                                             S13
   12  CONTINUE                                                          S14
       IF(BIG - 1.E-10)13,13,14                                          S15
   13  IF(IC.EQ.M) GO TO 20                                              S16
       IC = IC + 1                                                       S17
       GO TO 11                                                          S18
C      INTERCHANGE ROW IR WITH ROW CONTAINING LARGEST ELEMENT IN COL. IC S19
   14  IF(I.EQ.IR) GO TO 16                                              S20
       DO 15 L=IC,N                                                      S21
       TEMPA = A(I,L)                                                    S22
       A(I,L) = A(IR,L)                                                  S23
```

```
   15 A(IR,L) = TEMPA
C     PIVOT IS NOW A(IR,IC)
C     DIVIDE ROW IR BY PIVOT
   16 F = A(IR,IC)
      DO 17 L=IC,N
   17 A(IR,L) = A(IR,L)/F
C     REDUCE ELEMENTS ABOVE AND BELOW PIVOT TO ZERO
      DO 18 K=1,M
      IF(K.EQ.IR) GO TO 18
      IF(A(K,IC).EQ.0.) GO TO 18
      G = A(K,IC)
      DO 18 L=IC,N
      A(K,L) = A(K,L) - G*A(IR,L)
      IF(DABS(A(K,L)).LE.1.E-14)A(K,L) = 0.
   18 CONTINUE
      IF(IC.EQ.M) GO TO 21
      IC = IC + 1
   19 CONTINUE
   21 KER = 1
      RETURN
   20 KER = 2
      RETURN
      END
```

## BIBLIOGRAPHY


1. Thaler, G. J., Design of Feedback System, Dowden, Hutchingson and Ross, Inc., 1973.

2. Kuo, B. C., Theory and Applications of Step Motors, West, 1974.

3. Tou, J. T., Digital and Sampled-data Control Systems, McGraw-Hill, 1959.

4. Wightman, E. J., Instrumentation in Process Control, CRC Press, 1972.

5. Philco. Technological Center, Servomechanism Fundamentals and Experimentals, p. 195-200, 1964.

6. Bartee, T. C., Digital Computer Fundamentals, McGraw-Hill, 1972.

7. Digital Control Handbook, Digital Equipment Corporation, 1970.

8. Givone, D. D., Introduction to Switching Circuit Theory, McGraw-Hill, 1970.

9. NASA Contracter Report 617, Algebraic Method for Dynamic Systems, by G. J. Thaler, D. D. Siljak, and R. C. Dorf, November, 1966.

10. National Technical Information Service, Some Applications of Algebraic Method, by G. J. Thaler and D. D. Siljak, 1968.

## INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 2 |
| 2. | Library, Code 0212<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Professor George J. Thaler, Code 52Tr<br>Naval Postgraduate School<br>Monterey, California 93940 | 5 |
| 4. | Professor Alex Gerba, Jr.<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 5. | 1LT Yongyudha Nandidarbha, RTA<br>Military Research and Development Center<br>Supreme Command Headquarters, Ministry of Defense<br>Bangkok, Thailand | 1 |
| 6. | Library<br>Military Research and Development Center<br>Supreme Command Headquarters, Ministry of Defense<br>Bangkok, Thailand | 1 |
| 7. | Mr. Martin Dost<br>IBM<br>Monterey and Cottle Roads<br>San Jose, California 95114 | 1 |
| 8. | Mr. Chung Liu<br>IBM<br>Monterey and Cottle Roads<br>San Jose, California 95114 | 1 |
| 9. | Mr. Ben Fiorino<br>IBM<br>Monterey and Cottle Roads<br>San Jose, California 95114 | 1 |
| 10. | Mr. T.J. Cameron<br>IBM<br>Monterey and Cottle Roads<br>San Jose, California 95114 | 1 |

| | No. Copies |
|---|---|
| 11. Mr. Ed Vaughan<br>IBM<br>Monterey and Cottle Roads<br>San Jose, California 95114 | 1 |

Thesis
N2415
c.1

161351

Nandidarbha
An analysis of a digital positioning system.